TU MICRO

COMMODORE

N.° 6 · SEGUNDA EPOCA · 350 PTAS (IVA INCLUIDO)

LA EMOCION DE STARSHIP ANDROMEDA

TODO SOBRE FICHEROS RELATIVOS

ENFA IBERICA: INTERESANTES PERIFERICOS

MUSIC STUDIO

SONIDO PROFESIONAL PARA 64, 128 Y AMIGA



Me puedes regalar...

una raqueta, un ajedrez,
un jersey, una estilográfica,
o una radio para el coche...

pero lo que yo necesito...
es este Commodore 128.



Y lo necesito porque es el más 128 de todo el mercado. Porque es 3 ordenadores en uno solo, sólo con pulsar una tecla.

Lo necesito porque es compatible con el C-64, y todos sus programas.

Lo necesito, porque ya que me decido, quiero lo mejor.

Lo necesito, porque todavía hay un par de vídeo juegos en los que me gana mi vecino y eso no puede ser.

commodore



Microelectrónica y Control, s. a. *Unico representante de Commodore en España. c/ Valencia, 49-53 08015 Barcelona - c/ Ardemans, 8 28028 Madrid*

Director:
ANTONIO M. FERRER ABELLO
Redactor Jefe:
FERNANDO LOPEZ MARTINEZ
Redacción:
ANTONIO CARVAJAL
JOSE LUIS DE DIEGO
JUAN M. LOPEZ MARTINEZ
PABLO GARCIA MOLINA
IGNACIO BARCO LUENGO
ALFREDO SINDIN VALERO
FERNANDO ACERO MARTIN
Colaboradores:
JOSE LUIS M. VAZQUEZ DE PARGA
ANTONIO MANZANERA
Secretaria de Redacción:
PILAR MANZANERA AMARO
Maquetación:
CARLOS GONZALEZ AMEZUA
CARLOS TALLANTE
Ilustraciones:
ANTONIO PERERA
RAMON POLO
Fotografía:
EQUIPO GALATA
Directora Publicidad:
CARMINA FERRER
Publicidad Madrid:
BEGOÑA LLORENTE
Tel.: (91) 457 69 23
Publicidad Barcelona:
ISIDRO IGLESIAS
Avda. Corts Catalanes, 1010
Tel.: (93) 307 11 13
Director de Producción:
VICENTE ROBLES
Directora de Administración:
MARIA ANTONIA BUITRAGO
Suscripciones:
MARIA GONZALEZ AMEZUA
Redacción, administración, publicidad y suscripciones:
Plza. República Ecuador, 2.
28016 MADRID. Tel.: 250 58 20
Télex 49371 ELOC E
Dirección para correspondencia:
Aptdo. de Correos 61.294
28080 MADRID
TU MICRO COMMODORE es una publicación mensual de Ediciones INGELEK.

Fotomecánica:
RODACOLOR, S. A.
Impresión:
GRAFICAS REUNIDAS, S. A.
Distribución:
COEDIS, S. A.
Valencia, 245. Tel.: 215 70 97
08007 BARCELONA
Almacén: Nacional II. Km. 609,4
MOLINS DE REI (Barcelona)
Delegación en Madrid: Serrano, 165.
Tel.: 411 11 48
Almacén: Laforja, 19-21, esq. Hierro
Pol. Industrial Loeches
TORREJON DE ARDOZ (Madrid)

Precios para España. Ejemplar: 350 ptas. (IVA incluido). 330 ptas. (Canarias, Ceuta y Melilla).
(La suscripción anual incluye 11 números).
Distribución Cono Sur:
CADE, S.R.L.
Pasaje Sud América, 1532
Tel.: 21 24 64
Buenos Aires 1.290. Argentina.

Depósito Legal: M. 40920-1985.
Impreso en España

octubre 1986

Este mes, en nuestra sección de EQUIPOS, vamos a hablar de dos joystick poco habituales en lo que se refiere a lo que normalmente entendemos por joystick.

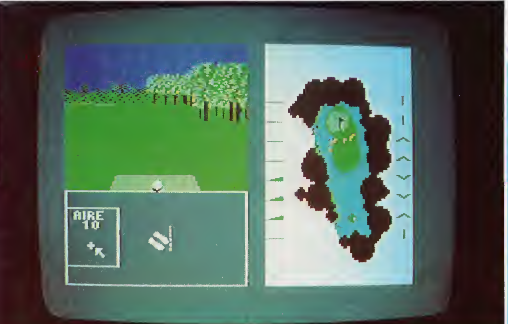
El joven coronel no daba crédito a sus oídos: se le ordenaba liberar al Universo de la mayor amenaza a que jamás se había visto sometida; y todo eso contando tan sólo con una STAR SHIP ANDROMEDA.



El sol cae de pleno sobre The Belfry, el viento es del noraeste con una velocidad de 10 Km/h, los greens están más duros de lo que podíamos esperar... y nosotros nos encontramos realmente jugando a GOLF CONSTRUCTION SET.

SUMARIO

Ya pasada la casi siempre brusca «reentrada en atmósfera» del mes de septiembre, nos enfrentamos a la entrada otoñal, que confiamos se haga más llevadera gracias a este TU MICRO COMMODORE 6, caracterizado por la gran variedad en la temática de sus artículos. Tendremos desde un artículo que trata el software en profundidad, hasta un interesante montaje hardware de aplicación robótica. Como no, nos acompañarán en el recorrido de nuestras 64 páginas, el comentario de los programas de actualidad, y el estudio de los periféricos, así como la programación desde los más diversos aspectos.



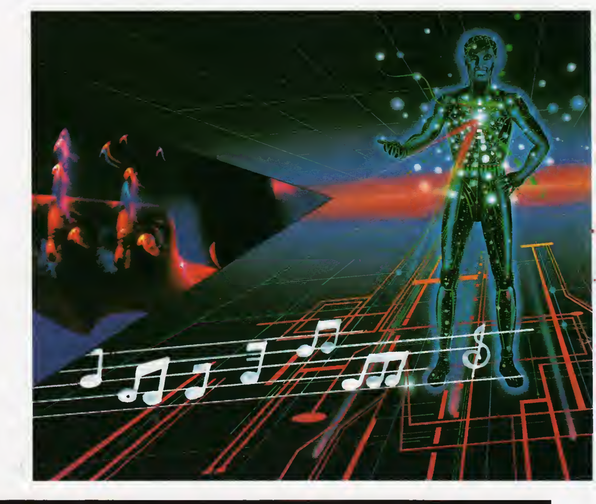
Novedad de Sistemas MIDI: Digidrum

Digidrum es un periférico que, conectado al C-64 ó al 128, permite transformarlo en una potente bateria electrónica programable de prestaciones y caracteristicas similares a las profesionales.

Tiene capacidad para memorizar un total de 50 patrones distintos y el correspondiente ensamblado de los mismos para formar un total de hasta 10 canciones.

Dispone de una libreria de más de 100 sonidos distintos, que pueden cargarse en bloques de ocho o individualmente. Se suministra con un manual de instrucciones, un diskette con el programa y una selección de sonidos, y un cartucho conectable al Commodore en el que se localiza la salida de audio y una de disparo externo.

El importador de este producto es Sistemas MIDI, sito en la calle Córcega, 89 de Barcelona, con teléfono 230 97 90. ●



Ergojoystick de First

El Joystick VG-125 construido en plástico duro de una larga duración y fiabilidad, dispone de cuatro botones de disparo (dos inferiores y dos superiores), una disposición anatómica para el ajuste de la mano, control fino de definición, ventosas inferiores para sujetarse mejor y poder dirigir el juego con auténtica precisión y libertad. Se puede emplear con los ordenadores Spectrum (con interface), Commodore 64 y 128, Vic 20, Amstrad 464/664/6128, Atari 400 y 800, Spectravideo, Thomson y sistemas MSX. Su precio 1.900 pesetas. ●

FISEX-86. Feria Informática Regional

La Universidad de Extremadura, a través del consejo de alumnos de la Facultad de Ciencias Económicas, ha organizado la III Feria de Informática y Servicios a la Empresa de Extremadura (FISEX-86), que tendrá lugar los días 15 al 19 de octubre, en las antiguas instalaciones de la factoría Itesa.

Fisex es una feria monográfica de ámbito regional que tiene como finalidad dar a conocer la situación del mercado de la informática y mecanización de oficinas, siendo uno de lo objetivos de esta nueva edición la ampliación de la feria hacia Portugal. ●

Enhorabuena a los ganadores

Este mes los agraciados ganadores de una suscripción por un año a nuestra revista TU MICRO COMMODORE, por su ayuda prestada en la confección de la sección LOS 7 MAGNIFICOS han sido los siguientes lectores:

Francisco Bernal Maillos, Cantalejo (Segovia).

Galo Sánchez Infante, Madrid.

Emma Oyarzabal, San Sebastián.

José Martínez Garrido, Navarcles (Barcelona).

Enric Manu, Espluges de Llobregat (Barcelona).

También en este apartado tenemo el honor de comunicar cuáles han sido los ganadores de la tercera edición del concurso de programación. Esta vez la merecida suerte ha recaido sobre:

1.er Premio: TERRORISTAS. Antonia González Fuentes.

2.º Premio: TEXT-64. Genaro Nieto Fernández. Villanueva y Geltru (Barcelona).

3.er Premio: TIRO AL PLATO. C.E.I.G. (Centre D'Estudis Informatics Ganxo). Sant Feliu de Guixols (Gerona).

Enhorabuena y ya sabéis, poneos lo más rápido posible en contacto con nosotros para gestionar vuestro premio. ●

Compulogical en la brecha

Compulogical ha aumentado su catálogo con la publicación de los últimos productos en el mercado de las firmas más representativas de software. Pasaremos a su enumeración:

Fairlight de Softek/Edge para C-64. Lo que hace de este juego algo especial, es la técnica de construcción en tres dimensiones.

Flight Deck CBM de Aackosoft para Commodore 64. La tercera guerra mundial está aquí: utiliza tu flota aero-naval.

Fiona de Leisure Promotions. Para C-64 nunca existió una bruja tan alocada como Fiona, cuando los cielos se oscurecen se convierte en demonio...

P.S.I. 5 de Accolade para Commodore 64. Avanza hacia el futuro, el siglo XXXV ya ha llegado.

Vorpal Utility kit de Epys para Commodore 64/128. Amplia las posibilidades de tu ordenador con la nueva tecnología Vorpal.

Y ahora pasaremos a detallar las novedades Epyx de estos meses:

– En verano:

Rogue para Commodore Amiga. Podemos pasar horas y horas en el mundo subterráneo de Rogue y no acabar de descubrir todos sus secretos.

Temple of Apshai Trilogy para Commodore 64/128 y Amiga. Tú eres el último aventurero en el templo de Apshai y debes sobrevivir para enfrentarte al Ra.

World Karate Championship para Commodore 64/128. Lucha hasta conseguir el cinturón negro mientras viajas por ocho países.

Super Cycle para Commodore 64/128. Debe alcanzar con tu moto por lo menos 230 km/h y resistir hasta el final de la carrera.

The Movie Monster Game para Commodore 64/128. ¿Por qué huir siempre de los monstruos?, no es mas fácil convertirte en uno de ellos y derrumbar la Torre Eiffel de un soplido.

– En septiembre:

Winter Games para Commodore Amiga.

World Games para Commodore 64/128. Sé uno de los atletas internacionales que compite en ocho pruebas alrededor del mundo. ●





Nuevo título de Anaya

Un nuevo título de Anaya disponible desde julio: Proyectos de música con microordenadores.

El propósito de este libro es mostrar algunos de los modos en los que se puede utilizar un ordenador doméstico, para la producción de música electrónica. Los temas cubiertos incluyen la secuencia y su control por medio de interfaces analógicos y MIDI, los ordenadores como líneas de delay digital, y generadores de sonido controlados por ordenador.

Este libro no está dirigido a los principiantes, pero por otra parte se ha mantenido a un nivel accesible, y no es necesario un excesivo conocimiento de hardware y de programación para construir y utilizar los circuitos descritos. ●

Starship Andromeda

El eco metálico de sus pasos resonaba por todo el pasillo cilíndrico en el que se encontraba. Llevaba aún puesto el uniforme de combate, ya que el Almirante en Jefe de la Junta Rebelde había requerido su presencia «a la mayor brevedad posible», según le informó el androide de mantenimiento del hangar B-78, en el que desembarcó tras unas maniobras con la escuadra que comandaba.

Llegó hasta la puerta que daba fin a aquel interminable pasillo. Los sensores de identificación actuaron con rapidez y la puerta se abrió automáticamente. Entró en la estancia. Al fondo, tras una enorme mesa, embutido en el típido traje negro con una banda dorada alrededor del cuello que indicaba su carga, estaba el máximo responsable de las fuerzas rebeldes, la última posibilidad de destruir la tiranía que la «Federación de mundos» había instaurado en toda la galaxia.

El Almirante dio media vuelta pesadamente, y miró con detenimiento al hombre que acababa de entrar. Su expresión, lejos de aparentar seriedad, reflejaba una hondísima preocupación. Su edad era indefinible, incómodamente indefinible. Invitó con un gesto al hombre a acercarse. Tenía la voz de quien ha vivido ya demasiado tiempo, pero su sonido aún era grave y solemne.

—La situación es desesperada, terriblemente desesperada. Andrac ha fracasado. Hemos perdido la astronave «Nebula» y Alona, según nuestros informadores, es cada vez más poderosa. Lo siento, comandante, son malas noticias, pero se trata de la realidad.

—¿Qué le ha ocurrido a Andrac? No se me informó de que estuviese realizando misión alguna.

—Lo sé, pero por razones de seguridad nadie, absolutamente nadie, aparte de los implicados directamente en la operación, tenían conocimiento de ella. Andrac fue enviado, como medida desesperada, con una pequeña dotación de hombres a reprogramar o destruir el SCM (Super Computador Mental) con el que Alona domina y somete a toda la galaxia. La última comunicación que recibimos de «Nebula» es extremadamente confusa, cada vez menos comprensible, y luego... el silencio.

—Ha muerto, entonces.

—O algo peor. Pero lo que importa en este momento es la situación presente. Ya no nos es posible combatir convencionalmente a las fuerzas de la Federación. Es demasiado tarde. Disponemos además de poco tiempo. Están a punto, si no lo han hecho ya, de descubrir la situación de nuestras bases. Pueden aniquilarnos a todos de un simple manotazo. La única posibilidad que tenemos es intentar de nuevo llevar a cabo la misión que Andrac no pudo terminar: atacar directamente al corazón de su poder, reprogramar el SCM.

—Pero eso no es posible. Aun considerando que se pudiese llegar hasta la base central de Alona, lo cual no es precisamente un juego de niños, no se puede reprogramar ese ordenador. Y no hablo sólo de la dificultad que esto supone, ya que no es ni muchísimo menos de tipo convencional, sino que no se puede acceder a él mas que empleando la llave de seguridad, que es imposible de conseguir. Esta llave es un cristal que Alona lleva siempre colgado del cuello mediante una cadena de Pentita. No se pueden fabricar copias del cristal, porque lo que el ordenador analiza es la información contenida en la estructura interna de éste, de lo que no sabemos nada. Usar el original es imposible, porque la Pentita es literalmente irrompible y... ¡Oh!, disculpe. Todo esto usted ya lo sabe.

—Sí, lo sé. Hemos analizado mucho esta situación y hemos llegado a la conclusión de que sólo puede llevarse a cabo la misión si se arrebata el cristal a Alona. En otras palabras: hay que romper la cadena.

—¿Y cómo se puede hacer?

—Con una lanza protónica.

—¡Por el gran Har'keath!, la lanza protónica es sólo una leyenda que se cuenta a los niños para dormir. Hace miles de años que fueron destruídas las 7 que había en la Galaxia, en los tiempos de la guerra con los Baarsens.

—Eso es lo que todo el mundo cree, pero antes de desaparecer, Andrac encontró una a bordo de un olvidado y viejo carguero espacial. Si pudiéramos conseguirla y hacerla funcionar...

FICHA TECNICA

NOMBRE	STARSHIP ANDROMEDA
PRECIO	2.500 PTAS.
SOPORTE	CINTA
TIPO	ACCION Y ESTRATEGIA
MODELO	C-64, C-128

OBSERVACIONES

¿DONDE ESTAN LAS INSTRUCCIONES?, MATA RILE RILE RILE

—Además, esa es otra cuestión. ¿Cómo funciona?

—Su manejo es el mismo que cualquier arma láser tipo Cyg-9000. Según las crónicas de la antigüedad, para hacerlo funcionar es necesario alimentarlo por una mezcla de iguales cantidades de Zyran y de Trysst. Hoy día es muy difícil encontrar estos elementos, pero sabemos de algunos pequeños planetas, más allá del sistema Riden, en los que se podría extraer la cantidad suficiente.

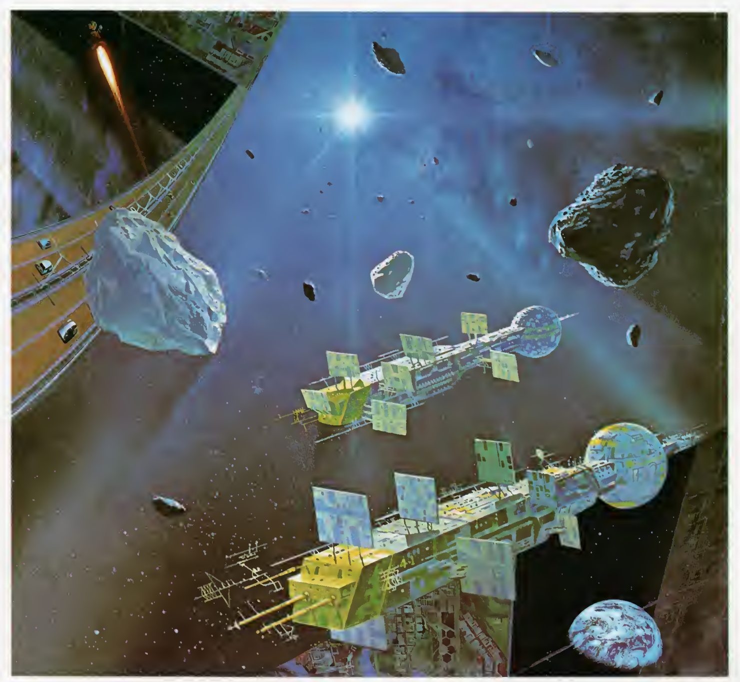
Se estará usted preguntando por qué le revelo esta información ultrasecreta. Seré sincero: quiero que sea el encargado de continuar la misión de su compañera Andrac. Esta es, por supuesto, una operación compuesta por voluntarios, y no le podemos obligar a nada. El consejo le ruega que acepte, dado que, con toda franqueza, es nuestra última oportunidad.

Cómo conseguir terminar con éxito con la federación y su tiranía

El joven coronel contempló a su superior largamente y en silencio, y descubrió en su mirada una lejana sombra de súplica. Comprendió que el destino de las fuerzas rebeldes, más aún, de la galaxia entera dependía de él, y comprendió también que no podía quedarse sin hacer nada, o al menos, sin intentarlo.

—Me honra la confianza que deposita el Consejo en mí. Iré. ¿De qué equipo dispongo?

—Gracias coronel, la Historia sabrá agradecérselo, al igual que a su compañera Andrac. Sobre el equipo hemos estado trabajando mucho. Tenemos reclutado una



dotación de 49 voluntarios estrictamente seleccionados. Irán a bordo de la astronave «Andrómeda».

—¿«Andrómeda»? No llegaríamos ni al sistema Kolex con una Mat-62 convencional como eso.

—Eso es lo queremos que piensen. Si lo conseguimos no se fijarán en ella, en algunos sectores no demasiado conflictivos, más que en una nave de transporte cualquiera. Pero hemos hecho algunos «pequeños» modificaciones que la transforman prácticamente en una Skp-45 de combate. Sus nuevos impulsores le permitirán alcanzar el metaespacio en pocos segundos, y va equipada con un potente láser no detectable por los scanners enemigos.

—De acuerdo, me parece una buena estrategia. ¿Se han modificado también los transmisores?

—Por supuesto. Hemos aumentado su radio de acción un 500% respecto a los iniciales. Por cierto, tengo que informarle sobre cuál será el método de utilización. No es posible indicarle, por razones de seguridad, todo su itinerario completo desde el principio. Una vez consiga terminar con éxito cada fase de la misión se le indicarán las coordenadas de su próximo destino. Así, si la transmisión es interceptada, la Federación no dispondrá de tiempo suficiente para anticiparse a sus movimientos, lo que le permitirá llegar al próximo destino sin problemas.

—¿Qué tipo de clave utilizaremos para descifrar las coordenadas?

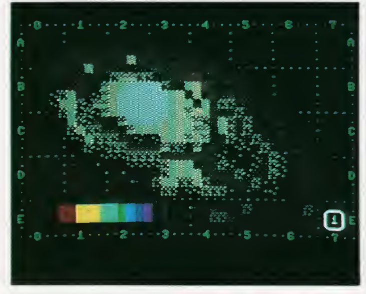
—Ninguna. Si realmente interceptan la comunicación, pensarán, igual que lo acaba de hacer usted, que las coordenadas están escritas en clave. Emplearán los descifradores que utilizamos (puedo asegurarle que los conocen todos), y enviarán naves a puntos equivocados. En realidad, sólo tendrá que hacer coincidir los cuatro números de las coordenadas, con los del sector del metaespacio en que se encuentre. Es demasiado simple como para que reparen en ello.

—Supongo que estará informado de mi historial completo. Han sido muy pocas las ocasiones en que he viajado por el metaespacio, no soy un piloto experimentado en este tipo de viajes.

—Sí, lo sé. Además, en los últimos meses nuestros científicos han descubierto algunas nuevas características de esta zona. Se encontrará tan sólo con tres tipos de objetos: estrellas azules, rojas y túneles espaciotemporales de salida. Las estrellas (no sabemos todavía bien el por qué), afectan a la









energía de la nave y al tiempo, aumentándolo o retardándolo, sólo por su proximidad. Los túneles deben ser tomados de frente, ya que si los impulsores salen del metaespacio antes que la cabina de mandos, la nave se desintegrará. Para los casos de colisión ha sido instalado un potente sistema de escudos deflectores, que consume mucha energía. Una vez salga del metoespacio, detenga la nave lo antes posible, para establecer contacto, ya que fuera del espacio real se crea una «zona muerta» en la que no se puede transmitir ni recibir información alguna.

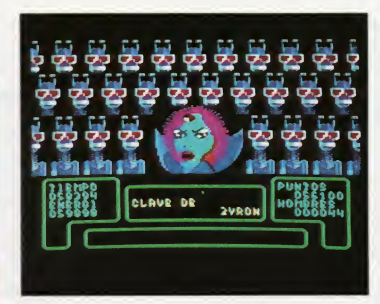
—¿Dónde se efectuará el primer contacto?

—Nada más partir. Una nave correo aliada, con un sofisticado sistema que la hace invisible, le indicará las coordenadas del planeta donde debe recoger Trysst para activar la lanza protónica. Detenga la «Andromeda» y busque con el scanner la zona sensitiva de la nave, para poder recibir la transmisión. Una vez llegue al planeta indicado, desciendo por las grutas hasta el fondo, recogiendo todo el mineral que encuentre. De regreso, recibirá el rumbo del siguiente planeta. En éste, las explotaciones de Zyron continúan por parte de la Federación, que ha instalado bases de defensa en la superficie. Deben ser destruídas mediante el láser adaptado a la computadora de a bordo.

El láser funciona automáticamente. Sobre su pantalla de ataque aparece un gráfico que puede desplazar a un lado y a otro, cambiando asimismo la perspectiva. Sitúe el gráfico sobre las bases, en la misma posición en que se encuentren, y el ordenador de disparo hará el resto. Desciendo luego al planeta y encuentre los depósitos de Zyron. Se le ha equipado con un detector de mineral, que produce un pitido que se vuelve más agudo conforme más cerca esté de la veta. Algunos depósitos son totalmente negros, ya que albergan un índice de radioactiviadad muy elevado. No los intente explotar. Utilice los escudos protectores para evitar los meteoritos. Cuando haya conseguido los cuatro depósitos viaje al siguiente planeta para recoger más Trysst.

—No parece demasiado arriesgado por el momento.

—Es cierto, pero una vez tenga todo el mineral debe acorralar a Alana en una de sus bases, para poderlo atacar. Entre en ellas defendiéndose de los cazas y apuntando la mira del láser a los círculos rojos de la sección giratoria. Dispone entonces de un determinado número de androides, que se









reproducen antes de cada movimiento. Los de Alana son los de la parte superior, y se mueven primero para ocupar una base. Haga lo mismo, con lo que se enfrentarán y desaparecerán, dándole ocasión a ocupar otra. Para operar en alguna de las tres bases debe primero dirigirse a ella, y volver a repetir el proceso de entrada.

—Una vez esté acorralada en una base tendré que conseguir la lanza protónica, supongo, pero, ¿dónde está?

VEREDICTO FINAL

GRAFICOS	* * * * *
SONIDO	* * * * *
ORIGINALIDAD	* * * *
DIFICULTAD	* * * * *
INTERES	* * * * *

—Se encuentra en el último de siete cargueros que encontrará. Estas naves tienen los comunicadores averiados, y al tratar de encontrar la zona sensitiva hay un 50% de probabilidades de que se destruyan. Entre en contacto y trate de encontrar la clave (un número de cuatro cifras) que le permitirá disponer de las coordenadas del siguiente carguero. La respuesta del comunicador será un porcentaje:

25% Número correcto en lugar correcto.
15% Número correcto en lugar erróneo.
0% Número erróneo.

La lanza protónica es invisible, y deberá utilizar el scanner para encontrarla. Una vez la tenga, diríjase a la nave nodriza enemiga y entre en ella. Dentro encontrará un puzzle que debe resolver, activando los fragmentos y cambiándolos con la lanza. Una vez resuelto, según nuestros informes, se podrá ver un plano de la nave principal de Alana. Una de las puertas parpadeará, indicando por dónde debe entrar. No tenemos completa seguridad, pero sospechamos que se pueda tratar, muy probablemente, de la situada más a la izquierda.

—¿Y una vez dentro...?

—Deberá encontrar la puerta de acceso a la sala del SCM, donde suele estar Alana. Deberá pasar una zona de sensores a los que disparará con la lanza de protones hasta que se vuelvan de color negro. No se acerque demasiado ni se quede quieto mucho tiempo, ya que le destruirían. Cuando consiga entrar, debe formar una clave de cuatro letras para reprogramar el SCM. Estas letras ya las debe haber visto antes: la forma de los depósitos de Zyron. No podemos informarle sobre el orden de éstas. Andrac intentó uno y fallo. El propio ordenador le destruyó.

Ahora ya lo sabe todo sobre la misión. Su nave está equipada y esperando para partir. Quiero desearle personalmente buena suerte y éxito hasta el final. Adiós.

—Adiós Almirante. Espero no fallarles.

Cuando la puerta se cerró, el líder de las fuerzas rebeldes asomó su mirada por un amplio visor, en el más completo silencio, hasta que vio partir la nave en las que estaban depositadas sus últimas esperanzas.

—Adiós, hijo mio, buena suerte...









La CIA 6526; El gran desconocido del C-64 (y II)

Este circuito está hecho a la medida por MOSTEK para COMMODORE. El aficionado sabe que lograr información sobre él es difícil (precisamente por esta característica «custom»). En este artículo vamos a explicar cómo funciona y para qué sirven sus registros.



ontinuamos este interesante tema, justo en el punta en que la abandonamas el mes pasado.

Temporizadores (TIMER A, TIMER B). Funcionamiento

Cada temparizadar cansiste en un cantadar de tiempa de sála lectura de 16 bits y en un registra de sála escritura de 16 bits (situados en las mismas direcciones). Las datas enviadas al temparizadar san almacenados en el registro de sólo escritura, cuando leemas el cantenida del temparizador, la que obtenemas es el cantenida del cantadar de tiempa de sála lectura. Los temporizadores se pueden utilizar independientemente o encadenados (mada link). Can este circuito es pasible lagrar la generación de retardas, pulsos de ancha variable, trenes de impulsas y frecuencias variables. Cada temparizador tiene asaciadas un registra de cantral, el cual actúa sobre las siguientes funcianes:

MARCHA/PARO

Este bit de cantral permite al microprocesador, activar a desactivar el temporizadar en cualquier momenta.

PB ACTIVADO/DESACTIVADO

Un bit de cantral, permite la aparición de la salida del temparizadar en una línea de salida del part B (PB6 para el timer A y PB7 para el timer B). Esta función sobrepasa el bit de control del DDRB, y fuerza a la apropiada línea del PB, a una salida.

PULSOS POSITIVOS Y NEGATIVOS/PULSO POSITIVO

Un bit de control, selecciona la salida aplicada al part B. Cada vez que se sobrepasa el valar seleccionado del temparizadar, se puede producir un tren de impulsas positivas y negativas a un pulsa positivo de un cicla de duración.

DISPARO SIMPLE/CONTINUO

Un bit de contral, selecciona entre dispara simple a continuo. En el mada de disparo simple, el temporizador cuenta desde el valor almacenado hasta cero, generando una interrupción que recarga el valor almacenada y se para. En moda cantinua, tras recargar el valar, repite el pracedimiento continuamente.

CARGA FORZADA

Un bit, permite almacenar el valor del registro del temparizador en el registra del cantadar de tiempo, esto puede hacerse en cualquier momento, este funcionando el temparizadar a no.

MODO DE ENTRADA

Un bit de cantral, permite la selección de la señal de relaj, empleada para decrementar el temparizadar. El timer A puede utilizar las pulsas del relaj 02 a pulsas externos aplicados al CNT. El timer B puede contar pulsas del relaj 02, pulsas externas del pin de CNT, así cama puede decrementarse por acarreos desde el temporizadar A, a par acarreas desde el temporizador A cuando la señal en el pin de CNT es positiva.

CIA # 1.

$DC04/56324 Temparizador A byte bajo.
$DC05/56325 Temparizador A byte alto.
$DC06/56326 Temparizadar B byte baja.
$DC07/56327 Temporizador B byte bajo.

CIA # 2.

Desde $DD04/56580 al $DD08/56584 realizan la misma función que las registras $DC04/56324 al $DC07/56327 de la CIA # 1.

RELOJ DIARIO

Este es un relaj diaria, para aplicacianes especiales en tiempo real, 24 haras (AM/PM). Este tiene una resalución de 1/10 de segunda y esta organizado en 4 registros: décimas de segunda, segundas, minutas y haras.

El indicadar AM/PM es el bit más significativo del registra de las haras, para que sea fácil su lectura. Cada registra se encuentra en formato BCD (decimal codificado en binario). El relaj requiere una señal externa de 50 a 60 ciclas, (pragramables par saftware), y tiene una alarma programable, la cual puede provocar una interrupción en el momento deseado, constan-

do su activación en el registro de interrupciones. Los registros de alarma son los mismos que los del reloj diario y están gobernados por un bit del registro de control; cuando este bit está activado, al escribir en los registros, lo que escribimos es la hora de la alarma en dichos registros (los registros de alarma son del tipo de sólo escritura).

Para poner en hora el reloj se tiene que seguir una secuencia determinada. El reloj en tiempo real, se para en el momento que se efectúa la escritura en el registro de las horas, y no comenzará a funcionar de nuevo hasta que no se escriba en el registro de décimas de segundo. Esto sirve para lograr que el reloj comience a contar en el momento deseado.

CIA #1.

$DC08/56328 Décimas de segundo.
$DC09/56329 Segundos.
$DC0A/56330 Minutos.
$DC0B/56331 Horas (bits 0-4), flag AM/PM (bit 7).

CIA #2.

En la CIA #2 las posiciones $DD08/56584 a $DD0B/56587 realizan las mismas funciones que en la CIA #1 las posiciones de $DC07/56328 a $DC0B/56331.

EL PORT SERIE

Este consiste en un registro de desplazamiento de 8 bits. Cuando son contados 8 pulsos en la entrada CNT, el byte contenido en el registro es enviado al bus paralelo y se genera una interrupción. La velocidad de transferencia se controla con el temporizador A. Para el envío de información se actúa a la inversa.

Como este sistema es bidireccional, se pueden conectar una gran cantidad de periféricos al bus serie, actuando el 5626 como controlador del bus, quedando otros dispositivos de la familia 65XX como esclavos, (hasta dos circuitos TTL) estos pueden ser:

6520 PIA (periferical interface adapter).
6522 VIA (versatile interface adapter).
6526 CIA (complex interface adapter).
CIA #1.
$DCØC/56332 Buffer de datos del bus síncrono serie.
CIA #2.
$DDØC/56588 Buffer de datos del bus síncrono serie.

CONTROL DE INTERRUPCIONES

Las interrupciones se pueden provocar en el C-64 por varios motivos:

a) Acarreo desde el temporizador A (en teoría cada sesentavo de segundo para IRQ).
b) Por el temporizador B.
c) Por la alarma del reloj de tiempo real de 24 horas.
d) Por el port serie.
e) Por la señal de —FLAG.

Un solo registro proporciona información sobre el enmascaramiento de las interrupciones. Es un registro doble, uno de solo lectura de datos y otro de solo escritura de máscaras.

CIA #1.

Se encarga de las interrupciones del tipo IRQ.

$DCØD/56333. Regitro de control de interrupciones. (Lectura de IRQ y escritura de la máscara de interrupciones). Bits:

7 Bandera IRQ. 1 = activada, 0 = desactivada.
4 Bandera de IRQ (lectura de cassette y entrada de SRQ del port serie).
3 Interrupción por port serie.
2 Interrupción por alarma de reloj diario.
1 Interrupción por el timer B.
0 Interrupción por el timer A.

ENMASCARAMIENTO DE INTERRUPCIONES

Para poder enmascarar una interrupción (permitirla o prohibirla), lo primero que debemos hacer es poner a 1 el bit más significativo del registro de enmascaramiento para poder modificar la máscara. Con vistas a permitir cualquier interrupción, el correspondiente bit tiene que estar a 1, ya que el otro modo queda desactivado.

REGISTROS DE CONTROL

El registro de control es un registro que permite modificar el modo de funcionamiento de la CIA; en cada una de ellas hay dos registros de control, cada uno de los cuales se encarga de controlar uno de los temporizadores.

CIA #1.



REGISTRO DE CONTROL TIMER A.
$DCØE/56334
Bits:

7 Frecuencia del relaj diaria 1=50 Hz. 0 = 60 Hz. (al encender el ordenodar o tras un reset se pane o 50 Hz.).

6 Mada del port serie 1 = salida 0 = entrodo.

5 Cantadar temparizador A. 1 = señal de CNT 0 = señol del reloj 02 del sistemo.

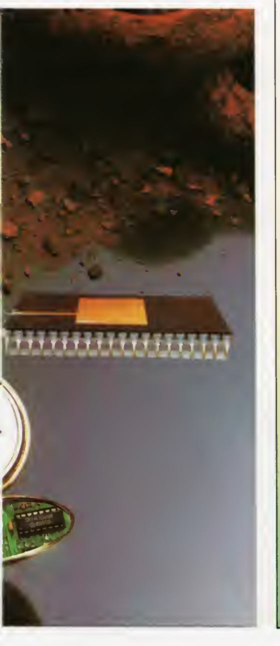
4 Fuerzo la carga del temporizodor A. 1 = si lo fuerza.

3 Mado de marcha del temporizodor A. 1 = pasa o poso. 0 = cantinua.

2 Modo de salido del temporizodor A a PB6. 0 = pulso 1 = tren de impulsas.

1 Salido del temparizadar A en PB6. 1 = si 0 = no.

0 Parada morcho del temporizadar A. 1 = morcha 0 = parada.



LISTADO

```
10                      ;********************************
20                      ;RELOJ 24 HORAS PERMANENTE
30                      ;TU MICRO COMMODORE
40                      ;F.A. ACERO 1986
50                      ;********************************
60                      .TITLE  'RELOJ 24 HORAS'
70                      .WIDTH  80
80                      .HEIGHT 66
90                      .SKIP   2
100                     ;VALORES CONSTANTES
110 VECIRQLO            -$0314
120 VECIRQHI            -$0315
130 NUEVAPOSLO          -$C01C
140 NUEVAPOSHI          -$C01D
150 LIBRE1              -$FE
160 LIBRE2              -$FD
170 CONTROLRCIA         -$DC0D
180 HORASTOD            -$DC0B
190 MINUTOSTOD          -$DC0A
200 SEGUNDOSTOD         -$DC09
210 DECIMASTOD          -$DC08
220                     .SKIP   2
230                     ;PROGRAMA DE RELOJ
240                     *-$C000            ;INICIO
250                     JMP DESCONEXION
260 CONEXION            SEI                ;DESACTIVA INTERUPCIONES
270                     LDA VECIRQLO
280                     STA NUEVAPOSLO
290                     LDA #$20           ;NUEVO SALTO IRQ LO
300                     STA VECIRQLO
310                     LDA VECIRQHI
320                     STA NUEVAPOSHI
330                     LDA #$C0           ;NUEVO SALTO IRQ HI
340                     STA VECIRQHI
350                     CLI                ;PERMITE INTERRUPCIONES IRQ
360                     RTS
370                     ;RUTINA DESACTIVACION DEL RELOJ
380                     .DBYTE  $EA31      ;ZONA ALMACENAMIENTO DEL VECTOR IRQ NOR
MAL
390 DESCONEXION         SEI
400                     LDA NUEVAPOSLO
410                     STA VECIRQLO
420                     LDA NUEVAPOSHI
430                     STA VECIRQHI
440                     CLI
450                     RTS
460                     ;RUTINA IRQ MODIFICADA
470                     LDA CONTROLRCIA
480                     ORA #%10000000
490                     STA CONTROLRCIA    ;ACTIVA 60 HZ
500                     LDA #$04
510                     STA LIBRE1         ;HI PANTALLA
520                     LDA #$10
530                     STA LIBRE2         ;LO PANTALLA
540                     LDY #$00           ,SET CONTADOR
550                     LDA HORASTOD
560                     PHA                ;ALMACENA HORA+ AM/PM
570                     AND #%01111111     ;ELIMINA AM/PM
580                     LDX #$BA           ;DOS PUNTOS VIDEO INVERSO
590                     JSR DECODE1
600                     LDA MINUTOSTOD
610                     JSR DECODE1
620                     LDA SEGUNDOSTOD
630                     LDX #$AE           ;PUNTO VIDEO INVERSO
640                     JSR DECODE1
650                     LDA DECIMASTOD
660                     JSR DECODE2
670                     PLA                ;SACA HORA+ AM/PM
680                     BPL ACTAM          ;SALTA SI AM ACTIVADO
690                     LDA #$90           ;P VIDEO INVERSO
700                     .BYTE   $2C
710 ACTAM               LDA #$81           ;A VIDEO INVERSO
720                     JSR PRINTOUT
730                     LDA #$8D           ;M VIDEO INVERSO
740                     STA ($FD),Y        ;SALIDA VIDEO
750                     LDA #$D8
760                     STA LIBRE1
770                     LDA #$01           ;CODIGO DE COLOR
780 LOOP1               STA ($FD),Y
790                     DEY
800                     BPL LOOP1
810                     JMP ($C01C)        ;SALTO NORMAL A IRQ
820                     ;DECODIFICACION BCD
830 DECODE1             PHA
840                     JSR DECODE3
850                     PLA
860                     JSR DECODE2
870                     TXA
880                     JSR PRINTOUT
890                     RTS
900 DECODE3             LSR A
910                     LSR A
920                     LSR A
930                     LSR A
940 DECODE2             AND #$0F
950                     ORA #$B0
960 PRINTOUT            STA ($FD),Y
970                     INY
980                     RTS
```

REGISTRO DE CONTROL DEL TIMER B
$DCØF/56335
Bits:

7 Ajustar alarma relaj diaria. 1 = alarma 0 = relaj.

6-5 Selecciát n del mada de temparizadar B. 00 = cuenta pulsas del relaj interna 02. 01 = cuenta transicianes pasitivas de CNT 10 = cuenta impulsas del temparizadar A. 11 = cuenta pulsas del temparizadar A mientras CNT sea pasitiva.

4-0 Cama el registra para el temparizadar A.

CIA # 2.

Esta CIA se encarga de las interrupcianes NMI (na enmascarables).

REGISTRO DE CONTROL DEL TIMER A
$DDØE/56590
Es cama el registra $DCØE/56334 de la CIA # 1.

REGISTRO DE CONTROL DEL TIMER B
$DDØF/56591
Es cama el registra $DCØF/56335 de la CIA # 1.

PROGRAMA EJEMPLO

A cantinuacián tenemas un pragrama que hace usa de las interrupcianes y el relaj en tiempa real de la CIA, para mantener un relaj de tiempa real a 60 ciclas mientras utilizamas el ardenadar.

Este permanecerá en la pantalla cantinuamente y na atrasará mientras na hagamas nada que vuelva a paner a 1 el bit de selección de frencuencia de aperación, cama RUN/STOP-RESTORE. Cama ejercicia pademas intentar panerle alarma a dicha relaj.

Esperamas que este ejempla sirva para aclarar las pacas dudas que hayan padida quedar, sabre la puesta en práctica de nuestras canacimientas a cerca de la CIA 6526, adquirirlas durante estas das últimas meses.

LISTADO

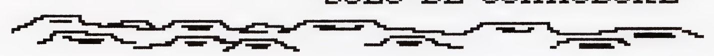
```
10 FORA=49152TO49300:READB:POKEA,B:NEXTA            -248-
20 PRINT"{CLR}{WHT}{RON}{6 ABJ}RELOJ C-64{ROF}"
                                                     -107-
30 PRINT:PRINT"ACTIVAR SYS 49263"                   -108-
40 PRINT:PRINT"DESACTIVAR SYS 49155"                -073-
50 SYS 49155                                         -172-
60 GOTO 500                                          -116-
100 DATA 76,30,192,120,173,20,3,141,28,192,169,45,141
,20,3,173                                            -025-
110 DATA 21,3,141,29,192,169,192,141,21,3,88,96,234,4
9,120,173                                            -046-
120 DATA 28,192,141,20,3,173,29,192,141,21,3,88,96,17
3,13,220                                             -242-
130 DATA 9,128,141,13,220,169,4,133,254,169,14,133,25
3,160,0,173                                          -130-
140 DATA 11,220,72,41,127,162,186,32,123,192,173,10,2
20,32,123,192                                        -215-
150 DATA 173,9,220,162,174,32,123,192,173,8,220,32,14
0,192,104,16                                         -177-
160 DATA 3,169,144,44,169,129,32,144,192,169,141,145,
253,169,216,133                                      -105-
170 DATA 254,169,1,145,253,136,16,251,108,28,192,72,3
2,136,192,104                                        -247-
180 DATA 32,140,192,138,32,144,192,96,74,74,74,74,41,
15,9,176                                             -016-
190 DATA 145,253,200,96,0                            -009-
500 PRINT:PRINT "{RON}PUESTA EN HORA"                -078-
510 POKE56334,PEEK(56334)OR128:POKE56335,PEEK(56335)A
ND127                                                -081-
520 INPUT"AM O PM";C$                                -231-
530 C=128:IFLEFT$(C$,1)="A"THENC=0                   -161-
540 INPUT"HORA";C$:GOSUB5000                         -216-
550 POKE56331,C+H                                    -205-
560 INPUT"MINUTOS";C$:GOSUB5000                      -223-
570 POKE56330,H                                      -096-
580 INPUT"SEGUNDOS";C$:GOSUB5000                     -026-
590 POKE56329,H                                      -106-
600 PRINT"CUANDO SEA EL MOMENTO PULSA UNA TECLA"
                                                     -092-
610 GETC$:IFC$=""THEN610                             -117-
620 POKE56328,0                                      -075-
630 PRINT"{CLR}":NEW                                 -065-
5000 IF LEN(C$)=1THENG=0:GOTO5020                    -182-
5010 I=VAL(LEFT$(C$,1))                              -004-
5020 K=VAL(RIGHT$(C$,1))                             -090-
5030 H=I*16+K:RETURN                                 -215-
```



COMPULAND

PRIMERA TIENDA EN ESPANA

SOLO DE COMMODORE



BIENVENIDOS COMMODORIANOS ALGUNOS NOS CONOCEREIS, OTROS NO.PARA TODOS OS VOY A DAR UNA BREVE INTRODUCION:
SOMOS UNA EMPRESA NACIDA CON EL DESEO DE HACER LLEGAR AL USUARIO,TODA UNA SERIE DE PRODUCTOS QUE O NO LLEGABAN A ESPANA O LO HACIAN CON MUCHO RETRASO.
VIAJAMOS CONTINUAMENTE A LOS MAS IMPORTANTES SHOWS TANTO EN INGLATERRA COMO EN ALEMANIA,NOS CONVERTIMOS EN IMPORTADORES Y EMPEZAMOS A DISTRIBUIR UNA SERIE DE ARTICULOS CONOCIDOS POR UNA MINORIA.
ACTUALMENTE ESTAMOS EN CONDICION DE OFRECER LAS ULTIMAS NOVEDADES DEL MERCADO NACIONAL Y EXTRANJERO

PREPARANDO LA CAMPANA DE NAVIDAD LOS PRIMEROS 100 PEDIDOS TENDRAN UN 20% DE DESCUENTO Y LOS SIGUIENTES UN 10%.
Toda persona que nos lo pida le enviaremos una lista con las ultimas novedades del mercado.Al mismo tiempo podeis enviarnos lo que a vuestro parecer son los 10 mejores juegos y las 5 mejores utilidades y lo que os gustaria que se sorteara entre los colaboradores.
A partir de Octubre empezara a funcionar la revista del Club,donde podeis participar.
No hay problemas de distancia por lo que para hacer pedidos o pertenecer al Club se puede hacer desde cualquier punto.



Para finalizar solo me resta deciros que ante todo somos usuarios y todo lo que de verdad sea bueno e interesante,intentaremos daroslo a conocer.Pero pensar que esta es una labor lenta pero continua y con la ayuda de todos vosotros podemos conseguir que todas esas maravillas aparecidas en revistas extranjeras las tengais disponibles en COMPULAND.
Poneros en contacto con nosotros para decirnos en que estais interesados,o para pedirnos informacion que lo antes posible os la enviare,ya que actualmente teneis disponible de todo aceleradores,backup,paquetes,digitalizadores etc mas de 1000 articulos.

Club de Soft

EL CLUB HA SIDO CREADO PARA DAR LA OPORTUNIDAD A LOS SOCIOS DE ACEDER A LA MAS EXTENSA Y COMPLETA BIBLIOTECA DE SOFT EN CINTA POR UNA MODICA CANTIDAD.
Condiciones:Compra de un programa y pago de 500 pts de inscripcion.No hay cuotas,pero si todos los meses compra de un programa.
Ventajas:Descuento del 10% en todos los productos.Sin limites para los cambios pagando 500 en cada uno.Siempre el ultimo programa que entre en tu poder es el primero en cambiar.Informacion periodica de todas las novedades y ....

Ahora que hemos demostrado ser los numeros uno en Commodore,vamos a volcarnos mas en utilidades y ayudas a la programacion que es un campo bastante abandonado.
A partir de Septiembre empezaremos a comercializar toda una serie de productos de total interes para cualquier usuario.
Nos vamos a ocupar tambien seriamente del 128 y AMIGA,sacando al mercado nacional desde digitalizadores de imagen,hojas de calculo,paquetes de gestion,graficos y todo lo que este disponible en el mercado internacional.



Control de motor

Dado que con las salidas del ULN2803 nos resulta imposible hacer girar un motor en dos sentidos, se nos hace imprescindible disponer de un circuito que nos permita el giro del motor en ambos sentidos.

Dicho circuito deberá ser sencilla (tanta en cancepción camo en realizacián), y par supuesta fiable. Y si a lo anterior añadimas el que tenga pocas camponentes, además nas resultará ecanómico y compacta. Así pues, éstas son las características de nuestro circuita.

Este nas permitirá hacer que un motor gire en dos sentidos dependiendo del estado lógica presente en sus entradas. Y par si esto fuera poca, inclusa puede permitirnos el variar su velocidad de farma relativamente sencilla, como veremos más adelante.

Con la versión ariginal del circuita podremos mover motares que cansuman corrientes de hasta 1 amperio. En caso de utilizar matores más potentes, sála será necesario sustituir los transistares puente par otros que sean capaces de manejar corrientes más elevadas.

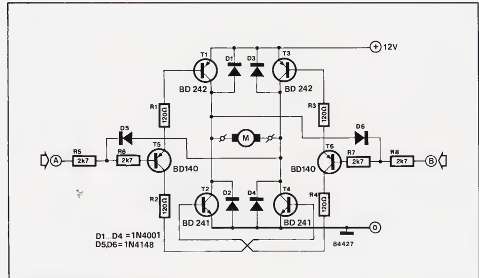
Cama camplemento al cantraladar de mator, incluímas la descripcián de una sencilla fuente de alimentacián, con la que podremos alimentar, tanto los motores como las tarjetas conectadas a nuestro BUS de entrada/salida.

Dicha fuente será capaz de proporcionarnos un voltaje simétrico de ± 12 voltios y una corriente máxima de 1 amperia. Par supuesto también nas praparciona las 5 voltios necesarios para hácer funcionar las tarjetas conectadas al BUS.

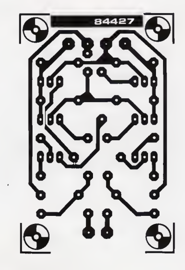
El conseguir valtajes distintas a éstas na representa más dificultad que el cambiar el transformadar y sustituir las circuitas reguladores de tensión.

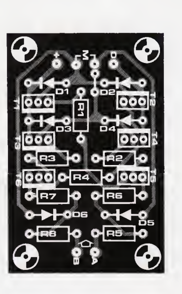
Como se puede observar en la figura, nuestro circuito de control posee dos entradas, cada una de las cuales irá canectada a una de las ocho salidas del interface.

Mientras que las dos entradas del circuita estén a nivel lógica alto no ocurre nada,



Esquemo del controlodor del motor.





Placa controlodor.

pero en el momento que alguna de ellas (por ejemplo la A) recibe un nivel de tensión próximo a cero voltios (cero lógico), el transistor de control T5 se satura, con lo cual los transistores puente T1 y T4 conducen, poniéndose el motor en marcha en un sentido u otro.

Cada pareja de transistores puente es controlada por un único transistor. Esto es así gracias a que cuando el transistor de control T5 pasa a saturación (conduce), T1 comienza también a conducir, provocando a su vez que comience a conducir su pareja (T4), entrando el motor en movimiento en un sentido u otro. En otras palabras: el transistor de control, dependiendo del estado lógico presente en su entrada, pone en conducción o en corte a los transistores puente T1 y T4.

Los diodos D5 y D6 se encargan de proteger el circuito en el caso de que ambas entradas A y B sean puestas a nivel lógico bajo al mismo tiempo.

Examinemos qué es lo que ocurriría en este caso.

La entrada A se pone a nivel lógico bajo, con lo cual los transistores T1 y T4 pasan a conducir. Entonces, el ánodo del diodo D6 está conectado al polo positivo, con lo cual fuerza un estado lógico alto en la entrada B, aunque dicha entrada esté a nivel lógico bajo mediante la salida del interface.

Por supuesto, lo anteriormente dicho también es válido para el caso contrario, forzando en este caso el diodo D5 un estado lógico alto en la entrada A.

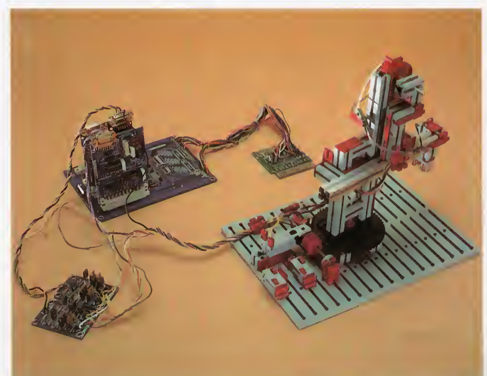
Si no existieran estos dos diodos de protección, los transistores puente quedarían destruidos al poner ambas entradas al nivel lógico bajo simultáneamente, puesto que los cuatro transistores puente estarían en conducción a la vez, pasando directamente la corriente del transistor T1 al T2, y del T3 al T4. Como en este caso no estaría por medio la resistencia eléctrica del motor, se produciría una sobreintensidad que destruiría los transistores puente.

Por último señalar que los diodos D1...D5 están encargados de proteger los transistores puente frente a las contracorrientes producidas al conectar el motor.

En el caso de que utilicemos un motor que consuma una corriente superior a 1 amperio, sustituiremos los transistores puente por otros que sean capaces de entregar más corriente.

Para corrientes de hasta 2 amperios podemos utilizar los BD243 en el lugar de los BD241, y los BD244 en el lugar de los BD242. Para corrientes todavía mayores sustituimos los BD241 por los BD643 ó 645, y para sustituir el BD242 utilizaremos los BD644 ó 645. Estos transistores son capaces de entregar una corriente de 4 amperios.





Fuente de alimentación

La fuente de alimentación descrita en la figura está en principio diseñada para suministrar una tensión simétrica de ±12 voltios, y una tercera adicional de 5 voltios.

La tensión simétrica nos servirá para alimentar, por ejemplo, los motores.

La tercera tensión de 5 voltios está pensada para alimentar todos los circuitos del BUS. Por supuesto, esta parte se puede omitir si alimentamos dichos circuitos mediante el micro.

Pero si por el contrario decidimos utilizar esta fuente de 5 voltios, deberemos desconectar la línea + del BUS de la línea +5 del micro, ya que si no lo hacemos estaríamos interconexionando las dos fuentes, y lo que podría pasar en este caso puede resultar fatal.

Descripción de la fuente

Al mirar el esquema lo primero que nos encontramos es el transformador. Como todos sabéis la misión del transformador es convertir una tensión en otra. En el caso que nos ocupa el transformador debe convertir los 220 ó 125 voltios de la red a los 15+15 voltios que precisamos. Por lo tanto necesitamos un transformador cuyo primario sea de 220 ó 125 voltios, según el voltaje de

nuestra casa, y un secundario que proparciane 15+15 voltias y 1 amperio.

Más adelante nos encantramos con los diodas D1...D4. Estas constituyen el puente rectificador, que convierte la tensión alterna en continua.

Después del puente están los condensadores C1 y C2. Son dos condensadores electrolíticos de elevada capacidad, cuya misión es filtrar la tensión procedente de la red, eliminando el zumbido de 50 Hz que ésta trae cansigo.

Los condensadores C3, C4, C5 y C6 tienen la misión de evitar que los circuitos integrados IC1 (7812) e IC2 (7912) entren en oscilación.

Dichos circuitos integrados son unos reguladores de tensión integrados de tres terminales. A su salida presentan un voltaje que puede cansiderarse constante.

Asimismo limitan la corriente máxima al valor de 1 amperia.

Estos circuitos integrados están protegidas contra sobrecargas térmicas (calentamiento excesivo), y cantra sobrecargas a su salida (cortocircuitos).

En definitiva, se puede decir que lo único que puede destruirlos es una colocación errónea.

Can respecto a IC3 (7805) decir que es un circuito integrado de la misma familia que las anteriares, con la única diferencia de que a su salida dispanemos de 5 voltios.

C7 y C8 son los encargadas de que IC3 no oscile, y por su parte R1 disipa parte de la potencia que es necesaria disipar para convertir 15 voltios en 5 voltios.

Para obtener otras tensiones de salida, camo ya habréis supuesto, sólo es necesario cambiar los integradas reguladores de tensión, y por supuesto el transfarmador.

Las dos primeras cifras que designan el tipo de integrada sirven para indicar si el regulador es para tensión pasitiva (78XX), o para tensión negativa (79XX).

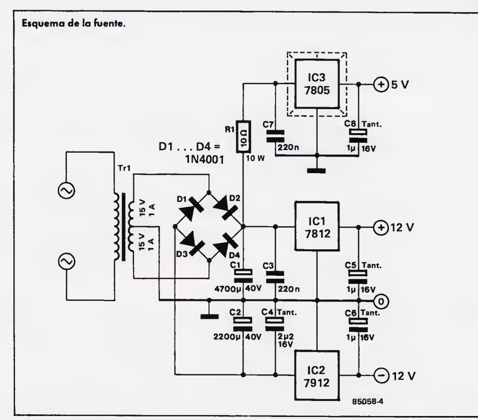
Las números que ocupan el lugar de las X designan el voltaje que dichos circuitos proporcionan en su salida.

Estos reguladores se fabrican para las siguientes tensianes de salida: 5, 6, 8, 10, 12, 15, 18 y 24 voltios.

El transformador deberá cambiarse, teniendo siempre en cuenta que debe proporcianar al menas 3 voltios más que la tensión de salida, excepto para los reguladores de 18 y 24 voltios que pueden trabajar con transformadores de igual tensión.

La tensión máxima de entrada para estos integrados es de 35 voltias excepto para el regulador de 24 V cuya tensión máxima de entrada san 40 voltios. Por otra parte, no debemos olvidarnos de cambiar también el valor de R1, calculando el nuevo valar mediante una simple regla de tres.

Esquema de la fuente.



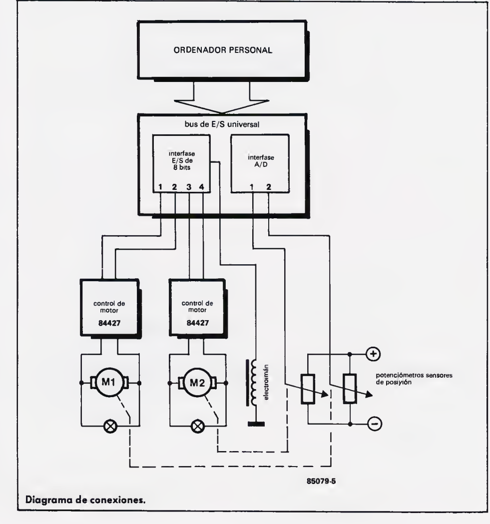
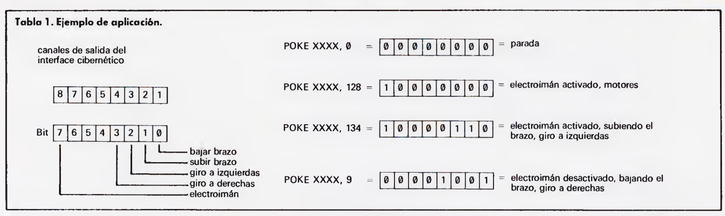
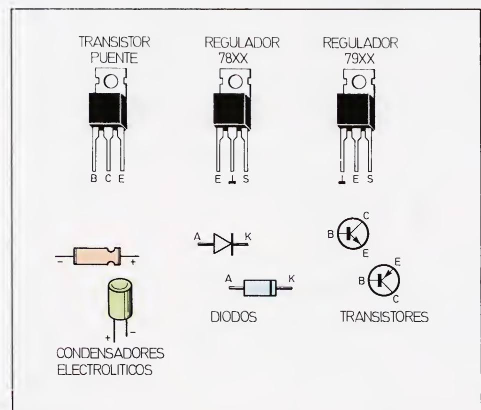


Diagrama de conexiones.

Tabla 1. Ejemplo de aplicación.





Montaje

Como de costumbre empezaremos soldando el único puente que existe en la placa. A continuación haremos lo propio con las resistencias, continuando con los diodos, teniendo siempre en cuenta que los diodos tienen polaridad. El cátodo está señalado por una franja de color cerca de uno de sus extremos. En el caso de los 1N4148 la franja es la de color amarillo. Tampoco deben calentarse en exceso, ya que se corre el peligro de destruirlos.

Ya por último soldaremos los transistores, que tampoco deben calentarse mucho. Haremos que coincida el trazo más grueso de la placa con la parte metálica del transistor.

Si observamos que los transistores se calientan en exceso durante su funcionamiento, será necesario dotarlos de un pequeño disipador metálico. Tengamos en cuenta que los disipadores o radiadores metálicos no pueden tocarse entre sí, ya que la parte metálica del transistor está unida eléctricamente con el colector, y si se juntan se dañarían los transistores.

Una vez terminado el montaje conectaremos el motor a sus terminales, y las entradas del controlador a dos salidas de interface. No olvidemos dejar sin conexión la patilla 10 del ULN2803.

Montaje de la fuente

Este se puede realizar sobre una placa universal tipo «uniprint». Dichas placas están perforadas de forma uniforme, y tienen unas pistas longitudinales.

Pues bien, haciendo cortes en dichas pistas y uniendo las que sean precisas mediante puentes, tendremos la placa lista para utilizar. Tomemos muy en cuenta el patillaje de los reguladores.

Comenzaremos por soldar la resistencia, continuando con los condensadores, teniendo cuidado en emplazar con la polaridad correcta los electrólicos y los de tántalo. A continuación soldaremos los diodos en su posición correcta así como los reguladores integrados de tensión. Al regulador de tensión IC3 es necesario añadirle un pequeño disipador.

Como alternativa a la utilización de la placa uniprint podemos diseñaros nuestra propia placa de circuito impreso, dibujando el trazado de las pistas sobre una placa virgen con un rotulador resistente al agua. A continuación atacaremos la placa ya sea con cloruro férrico o con atacador rápido. Todos estos productos los puedes encontrar en tu proveedor habitual.

Ya por último, después de realizar una última inspección, podemos darle tensión a los montajes.

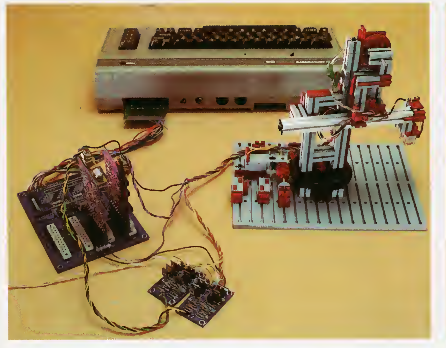
Utilización

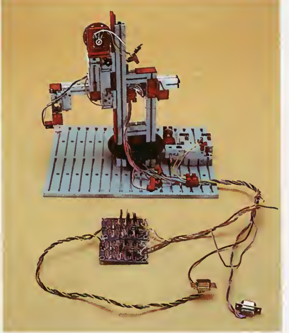
Como puedes observar en las fotografías, nos hemos ayudado de un conjunto Fischer-Technik para construirnos un pequeño brazo llamado torre de Hanoi.

Este brazo posee dos grados de libertad, con lo cual necesitaremos dos motores para moverlo. En el extremo del brazo lleva un electroimán de reducido tamaño con el que puede trasladar de un lugar a otro pequeños trozos de hierro.

Dado que hay que mover dos motores, hemos utilizado dos controladores de motor. Las entradas de dichos controladores están conectadas a las cuatro primeras salidas del interface.

Entonces, si el primer controlador está conectado a las salidas 1 y 2, cuando efectuemos un **POKE xxxxx,1** el motor se moverá en un sentido. Si ejecutamos un **POKE xxxxx,2** el motor se moverá en el sentido contrario. Para que se detenga, lógicamen-





te, habrá que realizar un **POKExxxxx0.** Para mayor claridad puedes guiarte por los ejemplos de figura.

El electroimán está conectado por una parte al polo positivo y por otra a la salida 8 del interface.

No olvidemos hacer el **POKE** a una de las cuatro direcciones ocupadas por el conector del BUS al que esté enchufado el interface.

La posición del brazo la sabemos mediante dos potenciómetros y el convertidor A/D.

En uno de los potenciómetros el eje con el armazón del brazo, y la carcasa del mismo es solidaria con la base sobre la que está montada el brazo. De esta manera, cuando el brazo gira, mueve el eje del potenciómetro, y mediante el convertidor analógico digital sabemos la posición de dicho eje, y por tanto, la del cuerpo del brazo.

El otro potenciómetro lleva solidario el eje a una rueda dentada que gira cuando el brazo sube o baja. La carcasa del potenciómetro es solidaria con el cuerpo del brazo. Y mediante el conversor A/D sabemos a qué altura se encuenta el mismo.

Como seguramente recordarás, para saber el contenido de las entradas del conversor A/D, hay que realizar un **POKE xxxxx,0:PEEK(xxxxx),** para la señal presente en la entrada 1, **POKExxxxx,1: PEEK(xxxxx)** para la señal presente en la entrada 2, y así sucesivamente.

La dirección a la que hay que realizar **POKE** y **PEEK** es cualquiera de las cuatro que tiene reservadas el conector al que enchufamos el convertidor.

Lista de componentes

CONTROLADOR:

Resistencias:
R1...R4 = 120 ohm. 1/2 w.
R5...R8 = 2k7 1/2 w.

Semiconductores:
D1...D4 = 1N4001
D5, D6 = 1N4148
T1, T3 = BD242
T2, T4 = BD241
T5, T6 = BD140

Varios:
Placa PCB84427

FUENTE

Resistencias:
R1 = 10 ohm. 10 w.

Condensadores:
C3, C7 = 220 nf
C5, C6, C8 = 1 microf./16 V. tántalo
C1, C2 = 2200 microf./40 V. electrolítico

Semiconductores:
D1...D4 = 1N4001
IC1 = 7812
IC2 = 7912
IC3 = 7805

Varios:
Transformador 15+15 V. 1 Amperio.
Placa uniprint.

Programa de control

Como todos sabemos, el programa de control será más complejo cuanto más complicada sea la aplicación que desarrollemos.

El programa que nosotros utilizamos para probar nuestros montajes es quizós el más sencillo que se pueda diseñar.

Este, lo primero que hace es saber en qué posición está el brazo. A continuación pide las nuevas coordenadas, y las compara con las actuales. Entonces dependiendo de si son mayores o menores que éstas, ordena a los motores que se pongan en marcha en uno u otro sentido.

En ese mismo instante se pasaró a una subrutina, a ser posible en código móquina, para dar más precisión, que compararó las posiciones por las que pasaró el brazo con las coordenadas introducidas. Entonces, en el momento en que la posición sea igual a la coordenada se parará el motor.

Por supuesto, puedes diseñar un programa tan complejo como quieras, al cual se le de una secuencia de operaciones, etc.

Lo que sí es recomendable es que la subrutina de comparación de posición se programa en código máquina para dar suficiente precisión.

La velocidad del motor la podemos variar haciendo que no esté siempre conectado. Es decir, poniéndolo en marcha y parando muy rápidamente. De esta manera, cuanto mayor sea el tiempo de desconexión con respecto al de conexión, el motor giraró mós lentamente.

Esta técnica de regulación de la velocidad se denomina control de velocidad por anchura de impulsos.



CON GRAN ENCICLOPEDIA INFORMATICA USTED PUEDE GANAR UN SENSACIONAL ORDENADOR PERSONAL EPSON PC (compatible con IBM PC) ADEMAS DE SU SISTEMA OPERATIVO CORRESPONDIENTE, EL PAQUETE OPEN ACCESS Y UN PROGRAMA DE CONTABILIDAD.



BOLETIN DE SUSCRIPCION

La colección GRAN ENCICLOPEDIA | INFORMATICA comprende 18 tomos de aparición quincenal. Las suscripciones empiezan a partir del nº 2 inclusive. (Si quiere también en nº 1 añada 495 ptas. al primer pago)

El pago de la suscripción lo puede realizar de la siguiente forma.

☐ APLAZADO: Contrareembolso de 9 plazos (9 envíos mensuales de 2 libros) de 1.800 ptas. El primer envío será de 1 tomo

☐ CONTADO: En un único plazo de 13.500 pesetas. Se realizarán 9 envíos mensuales de 2 tomos. El primer envío será de 1 libro. La forma de pago será la siguiente

☐ Talón bancario a nombre de INGELEK, S.A.
☐ Contra-reembolso del importe más gastos de envío

NOMBRE ______ EDAD ______

APELLIDOS ______

DOMICILIO ______

CIUDAD ______ PROVINCIA ______

C. POSTAL ______ TELEFONO ______ PROFESION ______

IMPRESCINDIBLE la firma paterna en los menores FIRMA

PRECIOS CON IVA INCLUIDO

Recorte o copie este cupón y mándelo a
EDICIONES INGELEK, S. A. Apartado de correos 61294. 28080 Madrid.

Tu Micropoly

Una revista es mucho más que una simple publicación, detrás se esconde todo un mundo de negocios, donde no sólo están permitidas «buenas» acciones. A ver que tal te desenvuelves en este terreno.

upongamos por un momento que nuestra querido publicoción TU MICRO COMMODORE, en lugar de pertenecer a Ediciones INGELEK, pertenece por partes iguales a uno, dos, tres o cuotro personos. Bien, sigamos suponiendo que estas cuatro personas compiten entre ellos para conseguir el control absoluto de lo publicoción. Poro ello, el gonodor deberó comprar cien millones de ejemplores de lo mismo, lo cuol supone un importante grovomen económico.

Las fuentes económicas al olconce de cado uno de los jugadores se obtienen en parte por suerte, en parte cuondo realizamos inversiones y en porte oprovechondo los inversiones de los demós. Otro mercodo que puede reportar grandes beneficios son lo compro (o 20 millones de pesetas cada uno) de ejemplores, combinodo con lo vento (o 25 millones de pesetas coda uno).

Para utilizor el programa no hoce folto ni siquiero joystick, sino simplemente teclear su lorgo listodo y después, con todo ilusión, dedicorse a jugor sobre el tablero. Todos los jugadores están representados sobre el toblero por un peón cuyo color es distinto de todos los demós. El movimiento sobre el to-

blero se efectúa con un dado simulado mediante la función aleatoria de los ordenadores Commodore 64 ó 128.

De vez en cuando la suerte nos será favorable y conseguiremos una puntuación de siete con este dado (que, por cierto, está algo trucado, porque los únicos dados capaces de ofrecer una puntuación de siete a un jugador están trucados). Bueno, cuando ocurre esto, habremos realizado sin ninguna intención una inversión extraña, el resultado de la misma es siempre impredecible, bueno o malo, pero impredecible.

El programa ofrece algo de animación a través de una serie de sonidos que ayudan a identificar el momento del juego en el que estamos. Por ejemplo, cuando captemos una inversión, ajena o propia, oiremos un sonido característico. Las inversiones son el apartado más importante de TU MICROPOLY.

Al caer sobre una casilla, donde otro jugador o nosotros mismos no estemos invirtiendo en dicho instante, se te ofrecerá la posibilidad de ingresar entre cero (no invertir) y nueve millones de dólares. Por contra, si caemos sobre una sección en la cual haya inversiones en curso, si son nuestras recibiremos el triple de lo invertido, pero si no, nos quedaremos con la inversión que allí haya. En dicho instante, la casilla ya está preparada para admitir una nueva inversión.

Es un juego simple, tranquilo, sin los derroches de adrenalina propios de las luchas alienígenas, pero ciertamente divertido.

LISTADO

```
1 DIM M$(16),N$(16),M(16)                                -251-
2 FOR N=1TO16:READ M$(N),N$(N),M(N):NEXT N               -225-
3 FOR N=1TO4:D(N)=100:NEXT N                             -191-
5 X(0)=40:Y(0)=198:X(1)=40:Y(1)=142:X(2)=40:Y(2)=86:X
(3)=115:Y(3)=86                                          -131-
6 X(4)=190:Y(4)=86:X(5)=255:Y(5)=86:X(6)=255:Y(6)=142
:X(7)=255:Y(7)=198                                       -079-
7 X(8)=190:Y(8)=198:X(9)=115:Y(9)=198                    -246-
10 FOR N=1TO67*3:READA:POKE49999+N,A:NEXT N              -150-
20 FOR N=1TO55*3:READA:POKE50999+N,A:NEXT N              -140-
30 FOR N=1TO55*3:READA:POKE51999+N,A:NEXT N              -142-
40 FOR N=0TO63:READA:POKE12288+N,A:NEXT N                -036-
50 FOR N=0TO63:READA:POKE12352+N,A:NEXT N                -029-
55 POKE 53269,0                                          -062-
60 POKE 2040,192:POKE 2041,192:POKE 2042,192:POKE 204
3,192:POKE 2044,193                                      -202-
70 POKE 53276,15                                         -111-
80 POKE 53285,0:POKE 53286,2:POKE 53287,1                -024-
90 POKE 53288,5:POKE 53289,10:POKE 53290,14:POKE 5329
1,0                                                      -106-
100 POKE 53256,165:POKE 53257,148:GOSUB 5000
                                                         -196-
110 POKE 53269,16                                        -157-
120 FOR N=54272TO54295:POKEN,0:NEXT N:POKE 54296,15
                                                         -020-
130 POKE 54277,9:POKE 54278,0                            -096-
140 FOR N=0 TO 66*3STEP3:POKE 54273,PEEK(50000+N):POK
E 54272,PEEK(50001+N)                                    -206-
150 POKE 54276,17:FORN1=1TO25*PEEK(50002+N)              -201-
160 GETW$:IFW$=" "THEN200                                -184-
170 NEXT N1:POKE 54276,16:NEXT N                         -161-
200 FORN=0TO3:POKE53269,0:FORN1=0TO150:NEXT N1:POKE 5
3269,16                                                  -123-
210 FORN1=0TO150:NEXTN1,N                                -049-
220 FORN=148TO0STEP-1:POKE53257,N:NEXTN:POKE53269,0
                                                         -174-
230 PRINT"{HOM}":PRINTTAB(12)"{BLK}{10 ABJ}NUMERO DE
JUG."                                                    -028-
240 PRINTTAB(12)"[2 ESP]1[2 ESP]2[2 ESP]3[2 ESP]4[2 E
SP]"                                                     -028-
250 FORN=54272TO54295:POKEN,0:NEXTN:POKE54296,15
                                                         -184-
260 POKE 54277,10:POKE 54278,23                          -193-
270 FOR N=0TO15:POKE 54274,N*15:POKE 54275,N:POKE 542
72,20:POKE 54273,30                                      -095-
280 GETW$:W=VAL(W$):IFW>0ANDW<5THENPOKE54276,64:GOTO3
10                                                       -190-
290 POKE 54276,65:NEXTN:POKE 54276,0                     -093-
300 GOTO 270                                             -165-
310 POKE 1595+(3*W),PEEK(1595+(3*W))+128                 -115-
320 FORN=0TO1000:NEXTN:GOSUB40000:E=50*W+100
                                                         -142-
325 POKE53248,40:POKE53249,198:POKE53250,40:POKE53251
,198                                                     -236-
327 POKE 53252,40:POKE53253,198:POKE53254,40:POKE5325
5,198                                                    -012-
328 Z=0:FORN=W-1TO0STEP-1:Z=Z+(2^N):NEXT N:POKE53269,
Z                                                        -182-
400 FOR A=1TOW:GOSUB 40000                               -050-
402 IF E(A)=>100 THEN 4000                               -123-
403 IF D(A)<0 THEN D(A)=0                                -090-
405 PRINT"{HOM}{PUR}{ABJ}{10 DCH}"RIGHT$("[6 ESP]"+ST
R$(D(A)),6);                                             -111-
407 PRINT"{17 DCH}"RIGHT$("[3 ESP]"+STR$(E(A)),3)
                                                         -223-
410 PRINT"{HOM}":PRINTTAB(14)"{PUR}{11 ABJ}JUGADOR ";
A                                                        -048-
420 FORN=54272TO54295:POKEN,0:NEXTN:POKE54296,15
                                                         -183-
430 POKE 54277,10:POKE 54278,23                          -192-
440 FORN1=1TO5:POKE 54272,20:POKE 54273,30:POKE 54276
,17:FOR N=0 TO 20:NEXT N                                 -233-
450 POKE 54276,17:POKE 54276,16                          -200-
460 FORN2=0TO100:NEXT N2,N1                              -134-
470 GOSUB 40000:POKE1563,32                              -024-
480 R=INT(RND(1)*7)+1:PRINT"{HOM}":PRINTTAB(17)"{PUR}
{F7 }{11 ABJ}";R                                         -207-
490 IFPEEK(197)=64THEN480                                -213-
500 IF R=7 THEN30000                                     -108-
510 P=P(A)+R:POKE 54272,9:POKE54273,0                    -094-
520 FOR N=P(A)TO P:Z=N                                   -093-
530 IF N>9 THEN Z=Z-10                                   -026-
540 POKE53246+(A*2),X(Z):POKE53247+(A*2),Y(Z)
```

```
                                                      -23S-
S50 POKES4272,40:POKES4273,22B:POKES4276,17:FORN1=1TO
300                                                   -252-
S5S NEXT N1:POKES4276,16:NEXT N                       -136-
S60 P(A)=P(A)+R                                       -0S7-
S65 IF P(A)>9 THEN P(A)=P(A)-10:O(A)=O(A)+10
                                                      -117-
S6B GOTO 20000                                        -014-
S70 NEXT A:GOTO 400                                   -131-
4000 GOSUB 40000:PRINT"{PUR}{HOM}{13 ABJ}{13 OCH} JUG
AOOR 1 "                                              -037-
4010 FOR N=S4272TOS429S:POKEN,0:NEXT N:POKE S4296,1S
                                                      -070-
4020 POKE S4277,9:POKE S427B,0                        -146-
4030 FOR N=0 TO 66*3STEP3:POKE S4273,PEEK(S0000+N):PO
KE S4272,PEEK(S0001+N)                                -000-
4040 POKE S4276,17:FORN1=1TO2S*PEEK(S0002+N)
                                                      -2S1-
40S0 NEXT N1:POKE S4276,16:NEXT N                     -210-
4060 GOSUB 40000:PRINT"{PUR}{HOM}{13 ABJ}{13 OCH} OTR
A[2 ESP]S/N "                                         -020-
4070 GETW$                                            -070-
40B0 IF W$="S" THEN RUN                               -07B-
4090 IF W$="N" THEN SYS 6473B                         -12B-
4100 GOTO 4070                                        -009-
S000 POKE S32B0,9:POKE S32B1,7:PRINT"{CLR}{BLK}";CHR$
(B);CHR$(142)                                         -161-
S010 PRINT" {ACL}[39 ESP]";                           -140-
S020 PRINT"{ACL}[2 ESP]{BLK}OINERO :[3 ESP]{PUR}100 $
{BLK}[2 ESP]EJEMPLARES :[2 ESP]{ACL} {PUR}0[4 ESP]";
                                                      -161-
S030 PRINT"{ACL}[2 ESP]{BLK}¦6 *¦{ACL}[12 ESP]{BLK}¦1
0 *¦{ACL}[B ESP]{GR1} {ACL} ";                        -019-
S040 PRINT"{ACL}[3B ESP]{BLK}[2 ESP]";                -031-
S050 PRINT"{ACL} {RON}{BLK}<BLK>{GRN}[7 ESP]{ROF}{ACL
} {RON}{BLK}<BLK>{CYN}[7 ESP]{ROF}{ACL} {RON}{BLK}<BL
K>{REO}[7 ESP]{ROF}{ACL} {RON}{BLK}<BLK>{GR3}[7 ESP]{
ROF}{ACL} {RON}{BLK}<BLK>{GR1} {ROF}{ACL} ";
                                                      -07B-
S060 PRINT"";                                         -247-
S070 PRINT"{ACL} {RON}{BLK} {GRN} TECLA {ROF}{ACL} {R
ON}{BLK} {CYN} SOFT[2 ESP]{ROF}{ACL} {RON}{BLK} {REO}
 EQUI[2 ESP]{ROF}{ACL} {RON}{BLK} {GR3} CURSO {ROF}{A
CL} {RON}{BLK} {GR1}O{ROF}{ACL} ";                    -247-
S0B0 PRINT"";                                         -249-
S090 PRINT"{ACL} {RON}{BLK} {GRN}[7 ESP]{ROF}{ACL} {R
ON}{BLK} {CYN}[7 ESP]{ROF}{ACL} {RON}{BLK} {REO}[7 ES
P]{ROF}{ACL} {RON}{BLK} {GR3}[7 ESP]{ROF}{ACL} {RON}{
BLK} {GR1}T{ROF}{ACL} ";                              -217-
S100 PRINT"";                                         -242-
S110 PRINT"{ACL} {RON}{BLK} {GRN} TECLA {ROF}{ACL} {R
ON}{BLK} {CYN}[2 ESP]WARE {ROF}{ACL} {RON}{BLK} {REO}
[3 ESP]POS {ROF}{ACL} {RON}{BLK} {GR3} BASIC {ROF}{AC
L} {RON}{BLK} {GR1}R{ROF}{YEL} ";                     -160-
S120 PRINT"";                                         -244-
S130 PRINT"{ACL} {RON}{BLK} {GRN}[7 ESP]{ROF}{ACL} {R
ON}{BLK} {CYN}[7 ESP]{ROF}{ACL} {RON}{BLK} {REO}[7 ES
P]{ROF}{ACL} {RON}{BLK} {GR3}[7 ESP]{ROF}{ACL} {RON}{
BLK} {GR1}A{ROF} ";                                   -039-
S140 PRINT"";                                         -246-
S150 PRINT"{BLK} {RON}[7 ESP]{ROF}<ROF>{ACL} {RON}{BL
K}[7 ESP]{ROF}<ROF>{ACL} {RON}{BLK}[7 ESP]{ROF}<ROF>{
ACL} {RON}{BLK}[7 ESP]{ROF}<ROF>{ACL} {RON}{BLK} {GR1
}S{ROF}{BLK} ";                                       -225-
S160 PRINT"{ACL}[13 ESP]{BLK}[14 ESP]{ACL}[10 ESP]{RO
N}{BLK} {GR1} {ROF}{ACL} ";                           -033-
S170 PRINT"{ACL} {RON}{BLK}<BLK>{NRJ}[7 ESP]{ROF}{ACL
} {RON}{BLK}<BLK>{PUR}[16 ESP]{ROF}{ACL} {RON}{BLK}<B
LK>{MRN}[7 ESP]{ROF}{ACL} {RON}{BLK} {GR1}I{ROF}{ACL}
 ";                                                   -216-
S1B0 PRINT"{ACL} {RON}{BLK} {NRJ}[3 ESP]A[3 ESP]{ROF}
{ACL} {RON}{BLK} {PUR}[16 ESP]{ROF}{CYN} {RON}{BLK} {
MRN}PROFE[2 ESP]{ROF}{ACL} {RON}{BLK} {GR1}N{ROF}{MRN
} ";                                                  -064-
S190 PRINT"{ACL} {RON}{BLK} {NRJ}[7 ESP]{ROF}{ACL} {R
ON}{BLK} {PUR}[16 ESP]{ROF}{ACL} {RON}{BLK} {MRN}[7 E
SP]{ROF}{ACL} {RON}{BLK} {GR1}V{ROF}{ACL} ";
                                                      -076-
S200 PRINT"{ACL} {RON}{BLK} {NRJ} FONOO {ROF}{ACL} {R
ON}{BLK} {PUR}[16 ESP]{ROF}{ACL} {RON}{BLK} {MRN} SIO
NAL{ROF}{ACL} {RON}{BLK} {GR1}E{ROF} ";               -117-
S210 PRINT"{ACL} {RON}{BLK} {NRJ}[7 ESP]{ROF}{ACL} {R
ON}{BLK} {PUR}[16 ESP]{ROF}{ACL} {RON}{BLK} {MRN}[7 E
SP]{ROF}{ACL} {RON}{BLK} {GR1}R{ROF}{NRJ} ";
                                                      -040-
S220 PRINT"{ACL} {RON}{BLK}[7 ESP]{ROF}<ROF>{ACL} {RO
N}{BLK}[16 ESP]{ROF}<ROF>{ACL} {RON}{BLK}[7 ESP]{ROF}
<ROF>{ACL} {RON}{BLK} {GR1}S{ROF}{ACL} ";             -044-
S230 PRINT"{ACL}[37 ESP]{RON}{BLK} {GR1}I{ROF}{ACL} "
;                                                     -030-
S240 PRINT"{ACL} {RON}{BLK}<BLK>{BLU}[7 ESP]{ROF}{ACL
} {RON}{BLK}<BLK>{RCL}[7 ESP]{ROF}{ACL} {RON}{BLK}<BL
K>{GR2}[7 ESP]{ROF}{ACL} {RON}{BLK}<BLK>{ACL}[7 ESP]{
ROF} {RON}{BLK} {GR1}O{ROF}{ACL} ";                   -206-
S2S0 PRINT"";                                         -24B-
S260 PRINT"{ACL} {RON}{BLK} {BLU} MICRO {ROF} {RON}{B
LK} {RCL} HARO[2 ESP]{ROF}{ACL} {RON}{BLK} {GR2}SOFT[
3 ESP]{ROF}{ACL} {RON}{BLK} {ACL}COOIGO {ROF} {RON}{B
LK} {GR1}N{ROF}{ACL} ";                               -0S9-
S270 PRINT"{GR3} {RON}{BLK} {BLU}[7 ESP]{ROF}{ACL} {R
ON}{BLK} {RCL}[7 ESP]{ROF}{ACL} {RON}{BLK} {GR2}[7 ES
P]{ROF}{ACL} {RON}{BLK} {ACL}[7 ESP]{ROF} {RON}{BLK}
{GR1}E{ROF}{ACL} ";                                   -164-
S280 PRINT"";                                         -2S1-
S290 PRINT"{ACL} {RON}{BLK} {BLU}[2 ESP]POLY {ROF}{AC
L} {RON}{BLK} {RCL}[2 ESP]WARE {ROF}{ACL} {RON}{BLK}
{GR2} MOOORE{ROF}{ACL} {RON}{BLK} {ACL}MAQUINA{ROF} {
RON}{BLK} {GR1}S{ROF}{ACL} ";                         -0BB-
S300 PRINT"";                                         -244-
S310 PRINT"{ACL} {RON}{BLK} {BLU}[7 ESP]{ROF}{ACL} {R
ON}{BLK} {RCL}[7 ESP]{ROF}{ACL} {RON}{BLK} {GR2}[7 ES
P]{ROF}{ACL} {RON}{BLK} {ACL}[7 ESP]{ROF} {RON}{BLK}
{GR1} {ROF}{ACL} ";                                   -121-
S320 PRINT"";                                         -246-
S330 PRINT"{ACL} {RON}{BLK}[7 ESP]{ROF}<ROF>{ACL} {RO
N}{BLK}[7 ESP]{ROF}<ROF>{ACL} {RON}{BLK}[7 ESP]{ROF}<
ROF>{ACL} {RON}{BLK}[7 ESP]{ROF}<ROF>{ACL} {RON}{BLK}
 {ROF}<ROF> ";                                        -170-
S340 PRINT"{ACL}[37 ESP]{HOM}";                       -069-
S350 POKE SS296,9:POKE SS33S,9:POKE S62S6,9:POKE S629
S,9                                                   -142-
S360 POKE 1024,10S:POKE 1063,9S:POKE 1984,128+95:POKE
 2023,12B+10S                                        -074-
S370 RETURN                                           -207-
20000 Z=0:IF P(A)<>0 THEN 20S00                       -1B0-
20010 PRINT"{PUR}{HOM}{12 ABJ}{12 OCH}TOTAL:";RIGHT$(
"[7 ESP]"+STR$(E),7)                                  -012-
20020 PRINT"{ARB}{12 OCH}1.[2 ESP]COMPRAR[2 ESP]"
                                                      -197-
20030 PRINT"{12 OCH}2.[2 ESP]VENDER[3 ESP]"           -006-
20040 GET W$:W1=VAL(W$):IF W1<>1 AND W1<>2 THEN 20040
                                                      -007-
20050 IF W1=2 THEN 20300                              -033-
20060 GOSUB 40000:PRINT"{HOM}{13 ABJ}{12 OCH}[2 ESP]C
ANTIDAD:[2 ESP]{IZQ}";                                -205-
20070 GET W$:W1=VAL(W$):IF W1<1 OR W1>9 THEN 20070
                                                      -104-
20075 IF Z=3 THEN S70                                 -162-
200B0 PRINTW1:W1=W1*20:IF O(A)<W1 OR (W1/20)>E THEN Z
=Z+1:GOTO 20060                                       -21B-
20090 PRINTTAB(13)"TOTAL:";RIGHT$("[9 ESP]"+STR$(W1),
6)                                                    -075-
20100 O(A)=O(A)-W1:E=E-(W1/20):E(A)=E(A)+(W1/20):GOTO
 S70                                                  -050-
20300 GOSUB 40000:PRINT"{HOM}{13 ABJ}{12 OCH}[2 ESP]C
ANTIDAD:[2 ESP]{IZQ}";                                -202-
20310 GET W$:W1=VAL(W$):IF W1<1 OR W1>9 THEN 20310
                                                      -098-
20315 IF Z=3 THEN S70                                 -159-
20320 PRINTW$:W1=W1*25:IF (W1/25)>E(A) THEN Z=Z+1:GOT
O 20300                                               -232-
20330 PRINTTAB(13)"TOTAL:";RIGHT$("[10 ESP]"+STR$(W1)
```

```
,6)                                                 -104-
20340 D(A)=D(A)+W1:E=E+(W1/25):E(A)=E(A)-(W1/25):GDTD
 570                                                -064-
20500 IF J(P(A))=0 THEN 20700                       -024-
20510 FDR N=54272 TD 54295:PDKE N,0:NEXT N:PDKE 54296
,13                                                 -215-
20520 POKE 54277,13:PDKE 54278,1                    -241-
20530 PDKE 54272,30:PDKE 54273,21:PDKE 54276,33
                                                    -060-
20540 FDR N=1 TD 30:NEXT N:PDKE 54276,32            -045-
20550 GDSUB 40000:PRINT"(HDM)(13 ABJ)(12 DCH) TDTAL[3
 ESP]:[2 ESP](IZQ)";                                -093-
20560 IF J(P(A))=A THEN H=G(P(A))*3:GOTD 20580
                                                    -036-
20570 H=G(P(A))                                     -029-
20580 PRINTRIGHT$("[2 ESP]"+STR$(H),2):D(A)=D(A)+H
                                                    -248-
20590 J(P(A))=0:G(P(A))=0:GOTD 570                  -090-
20700 GOSUB 40000:PRINT"(HDM)(13 ABJ)(12 DCH) INVERSI
DN:[2 ESP](IZQ)";                                   -051-
20710 GET W$:W1=VAL(W$):IF W1<1 OR W1>9 THEN IF W$<>"
0" THEN 20710                                       -241-
20720 IF W$="0" THEN 570                            -001-
20730 IF W1>D(A) THEN 20700                         -207-
20740 D(A)=D(A)-W1:PRINTW1:J(P(A))=A:G(P(A))=W1:GDTD
570                                                 -231-
30000 GOSUB 40000:PDKE 53264,2^(A-1):POKE53246+(2*A),
60                                                  -248-
30010 R=INT(RND(0)*16)+1:M$=M$(R):N$=N$(R):D(A)=D(A)+
M(R)                                                -105-
30020 PRINT"(HOM)(10 ABJ)":PRINTTAB(12);            -048-
30030 FOR N3=1TO13:PRINTMIO$(M$,N3,1);:GDSUB 41000:NE
XT N3                                               -200-
30035 PRINT:PRINTTAB(12);                           -053-
30040 FOR N3=1TD13:PRINTMID$(N$,N3,1);:GDSUB 41000:NE
XT N3:PRINT:PRINT"(ARB)"TAB(12)                     -184-
30060 IF M(R)>0 THEN 30100                          -137-
30070 FDR N=54272TD54295:PDKEN,0:NEXTN:PDKE 54296,10
                                                    -086-
30075 PDKE 54277,9:PDKE 54278,0                     -203-
30080 FOR N=0TD55*3STEP3:PDKE 54273,PEEK(51000+N):PDK
E 54272,PEEK(51001+N)                               -244-
30090 PDKE 54276,33:FOR N1=1TD40*PEEK(51002):NEXT N1
                                                    -234-
30095 PDKE 54276,0:NEXT N:GDTO 30200                -067-
30100 FDRN=54272TD54295:PDKEN,0:NEXT N:POKE 54296,10
                                                    -080-
30105 PDKE 54277,8:PDKE 54278,0                     -196-
30110 FDRN=0TD55*3STEP3:PDKE 54273,PEEK(52000+N):PDKE
 54272,PEEK(52001+N)                                -208-
30120 PDKE 54276,33:FOR N1=0TD40*PEEK(52002+N):NEXT N
1:PDKE 54276,0                                      -074-
30130 NEXT N                                        -196-
30200 POKE 53264,0:PDKE 53246+(2*A),X(P(A)):GDSUB 400
00:GDTO 570                                         -030-
40000 FDRN=0TO160STEP40:FDRN1=0TD15:POKE1475+N+N1,160
                                                    -198-
40010 POKE55747+N+N1,4:NEXTN1,N:RETURN              -095-
41000 FDR N=54272TO54295:POKEN,0:NEXTN:POKE 54296,15
                                                    -086-
41010 PDKE 54277,8:PDKE 54278,17                    -249-
41020 PDKE 54276,129:PDKE 54273,129:PDKE 54272,34
                                                    -172-
41030 PDKE 54273,28:PDKE 54272,124:PDKE 54276,128:RET
URN                                                 -196-
45000 DATA ARTICULD  VEN,DIDD.  30 $  ,30           -015-
45010 DATA FUERTE  CDMPE,TENCIA. -40 $,-40          -147-
45020 DATA PRIMICIA  EDI,TDRIAL.  50 $,50           -084-
45030 DATA MAQUETACIDN E,RRDNEA. -35 $,-35          -205-
45040 DATA PAPEL  MAS BA,RATD.  80 $  ,80           -221-
45050 DATA PERDIDA DE FD,TOS. -45 $   ,-45          -027-
45060 DATA MUY BUEN ARTI,CULD.  15 $  ,15           -055-
45070 DATA MUCHAS   ERRA,TAS. -10 $   ,-10          -243-
45080 DATA REVISTA   MAS,CARA.  35 $  ,65           -238-
45090 DATA IMPDRTANTE RE,TRASO. -55 $ ,-55          -215-
45100 DATA ATRACTIVDS DI,BUJDS.  40 $ ,40           -118-
45110 DATA FALTA DE MATE,RIAL. - 15 $ ,-15          -037-
45120 DATA GRAN REDACTDR,JEFE.  45 $  ,45           -040-
45130 DATA REDACTDRES VA,GDS. - 35 $  ,-35          -087-
45140 DATA UNA MAYDR  TI,RADA.  70 $  ,70           -241-
45150 DATA GRAVAMEN  FIS,CAL. - 60 $  ,-60          -012-
50000 DATA 51,97,1,51,97,1,68,149,1,68,149,1,86,105,1
,86,105,1,102,194,4                                 -022-
60010 DATA 86,105,2,102,194,1,102,194,1,115,88,1,102,
194,1,91,140,1,86,105,1                             -185-
50020 DATA 102,194,4,76,252,2                       -202-
50030 DATA 51,97,1,51,97,1,64,188,1,64,188,1,76,252,1
,76,252,1,91,140,4                                  -233-
50040 DATA 76,252,2                                 -227-
50050 DATA 102,194,1,115,88,1,102,194,1,91,140,1,86,1
05,1,76,252,1                                       -216-
50060 DATA 68,149,2,0,0,1                           -000-
50070 DATA 51,97,1,51,97,1,51,97,1,68,149,2,86,105,1,
51,97,1,51,97,1,51,97,1                             -231-
50080 DATA 68,149,4,86,105,2,68,149,1,68,149,1,68,149
,1,68,149,1                                         -173-
50090 DATA 64,188,1,68,149,1,76,252,2,0,0,1         -126-
50100 DATA 51,97,1,51,97,1,51,97,1,64,188,2,76,252,1,
51,97,1,51,97,1,51,97,1                             -226-
50110 DATA 64,188,4,76,252,2,102,194,1,115,88,1,102,1
94,1,91,140,1                                       -224-
50120 DATA 86,105,1,76,252,1,68,149,4               -094-
51000 DATA 51,97,2,51,97,2,68,149,2,68,149,2,64,188,6
,0,0,2                                              -155-
51010 DATA 45,198,2,45,198,2,57,172,2,57,172,4,51,97,
4,0,0,2                                             -197-
51020 DATA 43,52,2,57,172,3,51,97,1,51,97,2,45,198,6
                                                    -067-
51030 DATA 45,198,2,43,52,2,38,126,2,57,172,4,51,97,4
                                                    -114-
51040 DATA 86,105,4,76,252,2,68,149,2,64,188,2,57,172
,2,76,252,3,0,0,2                                   -178-
51050 DATA 76,252,2,68,149,2,57,172,2,57,172,2,51,97,
1,68,149,5,0,0,2                                    -133-
51060 DATA 86,105,1,68,149,1,68,149,1,51,97,1,51,97,1
,43,52,1,45,198,2                                   -191-
51070 DATA 76,252,6,0,0,2                           -001-
51080 DATA 76,252,2,57,172,2,64,188,2,76,252,2,68,149
,6,0,0,2                                            -253-
51090 DATA 76,252,2,86,105,3,86,105,2,68,149,4
                                                    -034-
52000 DATA 86,105,4,91,140,2,102,194,2,102,194,2,91,1
40,2,86,105,2,76,252,3                              -144-
52010 DATA 68,149,2,68,149,2,76,252,2,86,105,2,86,105
,4,76,252,4                                         -154-
52020 DATA 86,105,4,91,140,2,102,194,2,102,194,2,91,1
40,2,86,105,2,76,252,3                              -146-
52030 DATA 68,149,2,68,149,2,76,252,2,86,105,2,76,252
,4,68,149,4                                         -164-
52040 DATA 76,252,4,86,105,2,68,149,2,76,252,2,86,105
,1,102,194,1,86,105,2                               -120-
52050 DATA 68,149,2,76,252,2,86,105,1,102,194,1,86,10
5,2,76,252,2                                        -189-
52060 DATA 68,149,2,76,252,2,51,97,4                -055-
52070 DATA 86,105,4,91,140,2,102,194,2,102,194,2,91,1
40,2,86,105,2,76,252,3                              -151-
52080 DATA 68,149,2,68,149,2,76,252,2,86,105,2,76,252
,4,68,149,4                                         -169-
60000 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0          -228-
60010 DATA 0,0,0,0,0,0,0,0,0,2,0,0,2,0,0,2          -235-
60020 DATA 0,0,10,128,0,10,128,0,26,128,0,94,192,1,10
7,160                                               -058-
60030 DATA 5,106,160,1,106,160,0,26,128,0,5,0,0,0,0,0
                                                    -049-
60040 DATA 240,0,0,136,32,128,136,32,128,136,32,128,2
40,32,128,134                                       -218-
60050 DATA 56,134,129,36,137,133,36,137,135,56,102,0,
0,0,0,0                                             -174-
60060 DATA 0,0,1,128,0,224,64,1,17,64,1,1,192,1,0,0
                                                    -218-
60070 DATA 1,3,96,1,48,128,1,16,0,1,16,0,0,224,0,0
                                                    -174-
```



TU MICRO COMMODORE ha creado el revolucionario sistema de introducción de programas FUERA ERRORES. Este nos permitirá introducir, sin temor alguno al esfuerzo inútil, cualquier listado por largo y complicado que parezca.

Para adoptar los listados publicados bajo este sistema, deberemos seguir las siguientes normas:

1) Es fundamental transcribir EXACTAMENTE el listado reproducido, incluyendo todos sus espacios, aunque se trate de separaciones entre número de instrucción y línea de instrucción.

2) Todas las líneas finalizarán con un número de tres dígitos, encerrado entre guiones, que NO deberá ser introducido, puesto que no forma parte del programa, sino que tiene la finalidad de hacer funcionar el sistema FUERA ERRORES, según veremos más adelante. Para evitar equivocaciones, dicha cifra entre guiones se sitúa en el margen derecho del final de la línea BASIC a la cual corresponde, a una distancia prudencial del mismo.

3) Para facilitar la introducción de símbolos difícilmente interpretables, se procede a la siguiente representación en los listados.

— Las letras aparecidas entre menor y mayor deberán ser introducidas con pulsación simultánea de la tecla COMMODORE y la letra representada. Ej.: < M > =COMMODORE M.

— Las letras aparecidas entre barras verticales deberán ser introducidas como pulsación simultánea de la tecla SHIFT y la letra representada. Ej.:|K|= SHIFT K.

— Entre corchetes simples se representarán los símbolos que se obtienen por pulsación directa de la tecla, aunque lógicamente, este caso sólo se dará para indicar las sucesiones de más de una letra. Así por ejemplo, la introducción de 5 asteriscos se representaría por [5*].

— Para la repetición de símbolos obtenidos mediante las teclas COMMODORE o SHIFT, se seguirá una combinación de las tres normas anteriormente citadas. Así por ejemplo, la introducción de 10 símbolos COMMODORE H, se representaría por [< 10 H >].

— Para evitar confusiones, cuando se utilice el sistema de representación de sucesiones de carácter, y éste sea un espacio, se utilizará la abreviatura ESP. [15 ESP] = 15 espacios.

— Los caracteres de control, tales como desplazamientos del cursor, colores, estados de reversa y funciones, se simbolizarán por una abreviatura de tres letras (dos más un espacio en el caso de las funciones) encerradas entre llaves, tal como se señala en la tabla adjunta.

Para introducir cualquier listado por el sistema FUERA ERRORES, deberemos entrar previamente y ejecutar el listado BASIC que aparece en la página siguiente por lo cual es recomendable conservar una copia grabada del mismo, para sucesivas ocasiones. Una vez introducido este listado, ya sea por el teclado, o a través de cinta o disco, debemos ejecutarlo con **RUN**. Instantes más tarde aparecerá en la pantalla el mensaje **FUERA ERRORES!** y el cursor libre para la entrada de programas, con el tradicional **READY**. Por encima, lo cual indicará la activación del sistema de depuración de errores.

En virtud al **NEW** que finaliza línea 20 del programa ¡FUERA ERRORES!, éste habrá desaparecido de la memoria, y seremos libres para introducir cualquiera de los programas listados en la sección **TECLA A TECLA** de cualquier número de nuestra revista, o incluso aquellos que apareciendo en otras secciones se acojan a este sistema. Así pues, si el programa ¡FUERA ERRORES! ha desaparecido de la memoria, ¿qué hemos conseguido ejecutándolo? Bien, la respuesta se llama informáticamente **INTERRUPCION**: se trata de una técnica de programación en código máquina que permite que el ordenador efectúe prácticamente dos trabajos a un tiempo, o más correctamente, que ejecute determinada tarea de forma automática, sin necesidad de que le prestemos una atención constante, de forma similar al proceso de respiración en un humano.

Efectivamente, aunque el soporte BASIC ha desaparecido de la memoria, antes de «marcharse» ha dejado funcionando en modo interrupción la pequeña rutina en código máquina que se hallaba en sus **DATAS**. Para comprobarlo pulsa **RETURN**; observarás algo muy extraño: tu ordenador no se comporta normalmente, no sólo desciende una línea el cursor, sino que además hace aparecer un número en la esquina superior izquierda de la pantalla. A continuación veremos como emplearlo.

Cada vez que pulsemos **RETURN**, aparecerá un número en la mencionada zona de la pantalla, y éste corresponderá con la instrucción que hayamos introducido. Esto forma parte del sistema de FUERA ERRORES. Cuando introduzcamos cualquier línea de un listado de este tipo, deberemos fijarnos en el número que aparece al pulsar **RETURN** de fin de línea; si éste coincide con el que aparece en el listado al final de la línea, ésta habrá sido introducida correctamente, en caso contrario existe algún error de tecleado que debemos modificar. Para modificar una instrucción errónea, no tenemos ni tan siquiera que volver a teclearla si no queremos, bastará sencillamente con modificar el carácter o caracteres erróneos como siempre hacemos, hasta que coincida el número de verificación que se presentará al pulsar **RETURN**.

Así pues, el sistema FUERA ERRORES se compone de dos partes: una codificación especial de los listados que facilitan su introducción, evitando los errores al confundir los caracteres gráficos, de control, etc., y un sistema de verificación de líneas que nos advierte en el preciso instante de introducir una de estas, que está mal tecleada.

Ahora bien, el empleo de estos dos sistemas no quiere decir que nos encontremos ante un BASIC diferente al de COMMODORE 64. Este no ha cambiado, hace exactamente las mismas cosas de siempre; simplemente hemos cambiado la forma de hacer los listados. En cuanto al misterioso número que aparece en la esquina de la pantalla no es más que una simple suma de comprobación, lo que se conoce técnicamente como un CHECKSUM. La rutina en código máquina de interrupción suma los valores de los caracteres que entra-

TABLA DE INTERPRETACION DE CODIGOS DE CONTROL					
ABR	***SIFNIFICADO***	***OBTENCION***	***ABR***	***SIFNIFICADO***	***OBTENCION***
HOM	**HOME**	**CLR/HOME**	**F8**	**FUNCION 8**	**SHIFT F7**
CLR	**CLEAR + HOME**	**SHIFT CLR/HOME**	**BLK**	**BLACK (NEGRO)**	**CTRL 1**
ABJ	**CURSOR ABAJO**	**CRSR VERTICAL**	**WHT**	**WHITE (BLANCO)**	**CTRL 2**
ARB	**CURSOR ARRIBA**	**SHIFT CRSR**	**RED**	**RED (ROJO)**	**CTRL 3**
		VERTICAL	**CYN**	**CYAN (CIAN)**	**CTRL 4**
DCH	**CURSOR DERECHA**	**CRSR HORIZONTAL**	**PUR**	**PURPPLE (PURPURA)**	**CTRL 5**
IZQ	**CURSOR IZQUIERDA**	**SHIFT CSRS**	**GRN**	**GREEN (VERDE)**	**CTRL 6**
		HORIZONTAL	**BLU**	**BLUE (AZUL)**	**CTRL 7**
RON	**REVERSE ON**	**CTRL 9**	**YEL**	**YELLOW (AMARILLO)**	**CTRL 8**
ROF	**REVERSE OFF**	**CTRL 0**	**NRJ**	**NARANJA**	**COMMODORE 1**
F1	**FUNCION 1**	**F1**	**MRN**	**MARRON**	**COMMODORE 2**
F2	**FUNCION 2**	**SHIFT F1**	**RCL**	**ROJO CLARO**	**COMMODORE 3**
F3	**FUNCION 3**	**F3**	**GR1**	**GRIS 1**	**COMMODORE 4**
F4	**FUNCION 4**	**SHIFT F3**	**GR2**	**GRIS 2**	**COMMODORE 5**
F5	**FUNCION 5**	**F5**	**VCL**	**VERDE CLARO**	**COMMODORE 6**
F6	**FUNCION 6**	**SHIFT F5**	**ACL**	**AZUL CLARO**	**COMMODORE 7**
F7	**FUNCION 7**	**F7**	**GR3**	**GRIS 3**	**COMMODORE 8**

mos en cada línea que introducimos, aplicándoles un módulo 256, es decir, volviendo a cero cada vez que su suma parcial supera el 255, de este mndo se crea un número entre 0 y 255 dependiente directamente de los caracteres introducidos. Así pues, es prácticamente imposible que una sucesión de errores den por casualidad ese número, mientras que siempre que la línea esté correctamente introducida obtendremos el mismo código que nosotros hallamos por igual sistema y añadimos al final de cada línea al realizar el listado del programa.

Por tanto, el misterioso número no es tampoco una modificación del BASIC de COMMOOORE, sino simplemente un pequeño truco para la comprobación de que las líneas han sido bien introducidas. Propiamente no nos evita cometer errores de tecleado, sino que simplemente nos advierte inmediatamente en qué línea los hemos introducido.

Para desactivar el sistema sólo deberemos pulsar **RUN/STOP+RESTORE**, y si por cualquier motivo nos interesara reactivarlo, podríamos ejecutar **SYS 822**, siempre y cuando se encuentre el código máquina en la memoria, lógicamente.

¡ADVERTENCIA! Puesto que el código máquina se encuentra ubicadu en el bulfer del casete, es imprescindible desactivarlo **(RUN/STOP+RESTORE)** antes de realizar cualquier operación con dicho periférico.

Utilización del casete con fuera errores

Si queremos introducir parte de un programa, para continuar posteriormente el trabajo emprendido, sin perder por supuesto la enorme ventaja del FUERA ERRORES deberemos proceder de la siguiente forma:

1. Oesactivar el sistema FUERA ERRORES mediante la pulsación de RUN/STOP + RESTORE.

2. Trasladar el código máquina desde su ubicación en el buffer del casete hasta otro punto desncupadn en la memoria RAM (por ejemplo 49152). Para ello sólo hemos de saber que ocupa 114 bytes desde 822 (inclusive). Una sencilla rutina que efectúe el trabajo de reubicaciún puede ser: **FORI=0TO113:POKE49152+I,PEEK(822+I):NEXT.**

3. Realizar la operación cnrrespondiente con el casete; ya sea grabar una cupia de seguridad de lo introducidn en la memoria hasta el momento, ya sea cargar el casete una parte ya grabada del programa.

4 Restablecer FUERA ERRORES a su puntn original. Siguiendo el casn de la línea de ejemplu anterior. **FORI=0TO113:POKE822+I,PEEK(49152+I):NEXT**.

5. Reactivar el sistema FUERA ERRORES mediante SYS 822 (esta vez no aparecerá mensaje alguno, aunque al pulsar **RETURN** comprobaremos que aparecen las cifras de control en la esquina superior izquierda). Lógicamente, los pasos 4 y 5 no son necesarios si después de una grabación se va a apagar el ordenador, y sólo se llevarán a cabo después de las cargas, o si tras la grabación de una copia de seguridad del programa, deseamos continuar introduciéndolo acto seguido.

```
10 FORI=822TO935:READA:C=C+A:POKEI,A:NEXT
20 IFC<>15254THENPRINTCHR$(147)"ATENCION!, HAY UN ERROR EN LOS DATOS":END
30 PRINTCHR$(147)TAB(213)"FUERA ERRORES!":SYS822:NEW
100 DATA 169,3,141,37,3,169,69,141,36,3,169,0,133,254,96,32,87,241,133,251
110 DATA 134,252,132,253,8,201,13,240,13,24,101,254,133,254,165,251,166,252
120 DATA 164,253,40,96,169,13,32,210,255,165,214,141,176,3,206,176,3,169,0
130 DATA 133,216,169,18,32,210,255,169,19,32,210,255,169,45,32,210,255,166
140 DATA 254,224,100,176,5,169,48,32,210,255,224,10,176,5,169,48,32,210,255
150 DATA 169,0,133,254,32,205,189,169,45,32,210,255,173,176,3,133,214,76,88,3
```

CONCURSO



El travieso C-Byte tiene el honor de invitaros a la participación en nuestro cuarto concurso de programación. Los requisitos necesarios son bien pocos:

- Saber programar un ordenador COMMODORE.
- Ser español o extranjero y
- Tener una edad comprendida entre 5 y 105 años.

Fácil, ¿verdad?
En cuanto a los premios, la mar de atractivos:

- **1.er premio.—60.000 pesetas en material informático a escoger por el premiado, más un póster de C-Byte.**
- **2.° premio.—30.000 pesetas en material informático a escoger por el premiado, más un póster de C-Byte.**
- **3.er premio.—15.000 pesetas en material informático a escoger por el premiado, más un póster de C-Byte.**

Y en fin, si alguno de los programas destaca por su originalidad, estética o comicidad, no sería de extrañar que le cayera alguna cosilla más...

BASES DEL CONCURSO

1. Los programas remitidos al concurso deberán ser creación original del autor o autores, y completamente inéditos, pudiendo remitir tantos programas como se desee.

2. Los programas deberán ser enviados en casete o disco flexible a TU MICRO COMMODORE (Concurso de programación). Apartado de Correos 61.294. 28080 MADRID.

3. Los programas podrán ser de cualquier tipo (juegos, utilidades, gestión, educativos) y habrán de estar escritos en lenguaje BASIC o Código Máquina.

4. Los programas deberán ser remitidos desprovistos de cualquier tipo de protección, que impida o dificulte el análisis del mismo, así como reproducción en estas páginas y su introducción como listado siguiendo el sistema FUERA ERRORES.

5. Cuando la ejecución del programa precise de la concurrencia de determinado periférico o aditamento (joysticks, tabletas gráficas, programas comerciales de ayuda), se valorará la indicación de las modificaciones pertinentes, para que el programa pueda ser disfrutado por cualquier usuario en la configuración básica.

6. Todo programa presentado al concurso deberá acompañarse de los siguientes datos.

- Datos personales del concursante.
- Nombre del programa.
- Modelo para el que está destinado.
- Breve descripción del programa detallando las indicaciones necesarias para su ejecución.

7. Los programas premiados pasarán a ser propiedad de la revista TU MICRO COMMODORE, pudiendo hacer ésta libre uso de ellos, y renunciando sus autores a cualquier otra compensación distinta al premio.

8. Los programas no premiados que por su calidad se hagan merecedores de su publicación, serán adquiridos por la editorial, aplicando la tarifa vigente.

9. Los programas recibidos con posterioridad a la fecha tope de la presente edición, serán automáticamente incluidos en los destinados a la siguiente.

10. El jurado decidirá sobre todos los aspectos no contemplados en estas bases y su decisión será inapelable.

Y ahora a darse mucha prisa, el plazo para la recepción de programas termina el próximo día 1 de ____________

¡¡SUERTE!!

CURSO DE BASIC

ESTRUCTURA DE LOS DATOS

En los capítulos anteriores hemos visto, de forma general, cómo el ordenador es capaz de tratar diferentes tipos de datos, para realizar con ellos operaciones de cálculo (en el caso de los numéricos), fragmentación (en el caso de los alfanuméricos) o comparación (en ambos casos). Lo que no hemos hecho, hasta ahora, ha sido penetrar en la estructura interna de los mismos, describiendo la forma en que estos son almacenados en el interior del ordenador.

Simplificando mucho las cosas, podemos suponer que la memoria del ordenador es como una gran hoja de papel cuadriculado, sobre la cual podemos escribir y reescribir tantas veces como deseemos. Aún más, podemos elegir exactamente el lugar donde deseamos hacerlo, indicando el número correspondiente a una casilla determinada, siendo realmente ésta nuestra única restricción, puesto que nunca podremos escribir «a caballo» entre dos de ellas.

Al conjunto de esta memoria se le denomina **RAM (Random Access Memory),** cuya traducción es: Memoria de Acceso Aleatorio o directo. Con ello queremos decir que tenemos acceso a cualquier parte de ella, ya sea para conocer su contenido o para alterarlo, reemplazándolo por uno nuevo, con sólo reseñar su posición absoluta dentro de la misma.

Con una estructura similar a la descrita, el fabricante ha incluído también dentro de nuestro ordenador otra «hoja de papel cuadriculado», aunque ingeniándoselas de forma que podamos conocer su contenido pero no alterarlo. De hecho, podemos tratar de escribir sobre ella, siendo inútil nuestro esfuerzo por destruirla.

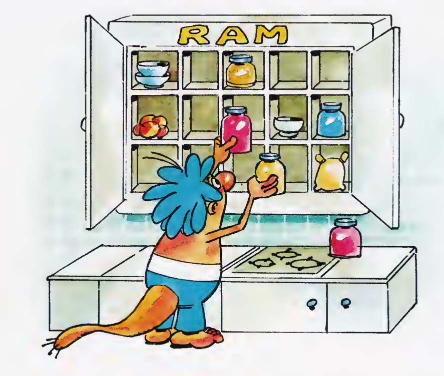
A esta memoria de sólo lectura se le denomina **ROM (Read Only Memory),** cuya traducción al castellano sería: Memoria de Sólo Lectura. El motivo de la protección especial de que dispone es evidente, puesto que contiene los elementos esenciales de programación de que dispone el ordenador para hacerlo trabajar: su sistema operativo, el lenguaje de programación, etc...

En total, el COMMODORE 64 dispone de 65536 posiciones de memoria o cuadrículas, pertenecientes a uno u otro tipo de las memorias descritas; y en cada una de ellas puede almacenarse un carácter alfabético, numérico o símbolo. A cada una de estas posiciones descritas se le denomina **BYTE**

La memoria del ordenador es como una gran hoja de papel cuadriculado sobre la cual podemos escribir cuantas veces deseemos.



La memoria RAM se caracteriza por tener acceso a cualquier parte de ella, ya sea para conocer su contenido o para alterarlo.



u «octeto». Realmente, el nombre de octeto es el más descriptivo para nosotros, pues indica que cada una de las posiciones de memoria está compuesta a su vez por un conjunto de ocho unidades elementales, denominadas **BITs.**

Un bit no es más que un impulso elemental binario, el cual puede adoptar dos únicos valores, a los cuales solemos asignar un significado lógico o simbólico: SI o NO, VERDADERO o FALSO, o 1 y 0. Aunque esta estructura puede parecernos en principio difícil de comprender, diremos que es, sin embargo, la única aceptable internamente por el ordenador, el cual no deja de ser un dispositivo electrónico que responde únicamente a impulsos eléctricos.

En cualquier caso, no debe preocuparnos la estructura interna de almacenamiento en la memoria del ordenador más que a nivel de posición de memoria (byte), por ser ésta la mínima cantidad de información con sentido para nosotros, al ser capaz de albergar un carácter.

La memoria de sólo lectura se denomina ROM.



LAS BASES DE NUMERACION

Es necesario que refresquemos ahora algunos conocimientos básicos sobre la aritmética binaria. Para empezar diremos que, del mismo modo que un dígito en base 10 (decimal) puede admitir un valor de 0 a 9, en base 8 (octal) puede albergar un número de 0 a 7, y en base 2 (binaria) únicamente los dígitos 0 y 1. Esto equivale a decir que, en cualquier base de numeración, los dígitos disponibles van desde el 0 hasta la base menos 1.

Si en base binaria sólo disponemos del 0 y el 1, en dos dígitos podemos albergar un número no superior a 3, en tres uno no superior a 7, en cuatro uno no superior a 15... y en ocho un número que se encuentre en el rango de 0 a 255. Generalizando, podemos decir que siendo **N** el número de dígitos y **B** la base, se cumple la fórmula: **X=B ↑ N−1.** Donde **X** indica el mayor número que podemos almacenar.

Particularizando esta fórmula para **B=2** y para **N=8,** obtenemos el resultado de **X=255.** Este es precisamente el caso que nos ocupa, puesto que en un byte u octeto de memoria, podemos almacenar un número comprendido entre 0 y 255. Esta cifra (256 posibilidades incluyendo el cero) se corresponde con las dimensiones del juego de caracteres del ordenador, es decir, el conjunto de todas las letras, números y símbolos que encontramos en el teclado, e incluso algunos más, aunque no representables en pantalla.

EL JUEGO DE CARACTERES

Como ya sabemos, al ordenador le resulta mucho más fácil llevar a cabo el tratamiento de números que no de le-

El Código ASCII, abreviatura de American Standard Code for Information Interchange, es un código de extraordinaria difusión compuesto por 128 caracteres.

EL CODIGO ASCII

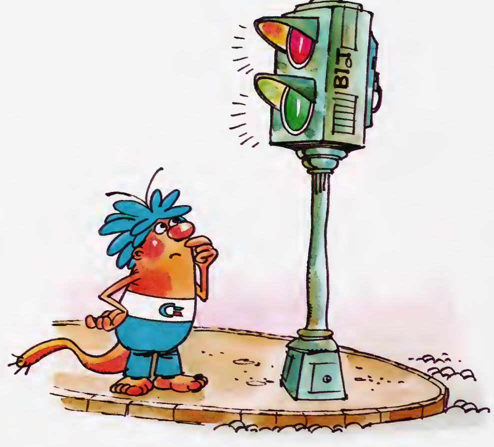
BITS 3210	BITS 654 000	001	010	011	100	101	110	111
0000	NUL	DLE	SP	0		P	£	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	´	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	v
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	F	,	<	L	/	l	|
1101	CR	GS	-	=	M	]	m	}
1110	SO	RS	.	>	N	i	n	ñ
1111	SI	US	/	?	O	—	o	DEL

tras, y lo que es más, en realidad sólo puede internamente efectuar un tratamiento numérico muy restringido, a base de unos y ceros. Sin embargo, a través de determinados sistemas de codificación es capaz de llevar a cabo complejas operaciones matemáticas.

Por ello, el ordenador codifica los caracteres en forma de números, lo cual le permite dar el tratamiento adecuado a los caracteres, ya sean letras, símbolos, etc... Por otra parte, este proceso de codificación interna no es más que una adecuación a un medio que nos es bastante familiar.

Así, cuando Samuel Morse inventó su conocido código telegráfico, lo hizo obligado por la necesidad de transmitir únicamente impulsos eléctricos de dos longitudes diferentes (puntos y rayas). Otra codificación similar, esta vez visual, es la que siguen las banderas de señalización en los barcos. Así pues, el hecho de codificar los caracteres por otro sistema, fácil de manejar con las herramientas de que se dispone, no es precisamente una de las innovaciones tecnológicas aportadas por los ordenadores. En todo caso, vamos a estudiar la codificación de caracteres que lleva a cabo nuestro equipo.

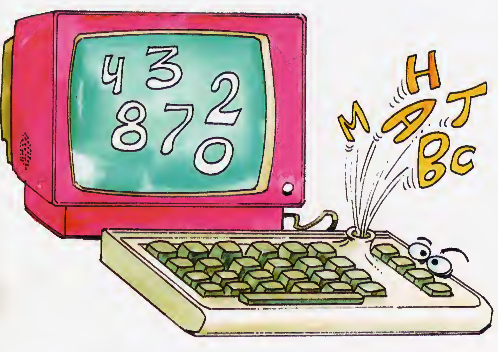
Un bit no es más que un impulso elemental binario, el cual puede adoptar dos únicos valores.



Desde la aparición de los ordenadores se hizo necesaria la creación de un código estandarizado, de cara a facilitar la comunicación entre los diversos aparatos. De esta idea, surgió el estándar de más frecuente utilización: el código ASCII. Estas siglas corresponden a **American Standard Code for Information Interchange,** que podríamos traducir como «Código Normalizado Americano para el Intercambio de Información». El mencionado código está compuesto por 128 caracteres, numerados del 0 al 127, que se detallan en el cuadro de más arriba.

Como podemos ver, este incluye tanto letras mayúsculas como minúsculas, símbolos, signos de puntuación y caracteres de control; en definitiva, todos aquellos caracteres que pueden ser necesarios para cualquier comunicación. Sin embargo, aún quedan libres 128 códigos (del 128 al 255). Estos códigos son particulares de cada ordenador, escapando por lo tanto a la estandarización que el código ASCII supone.

Así pues, cada fabricante de orde-



Al ordenador le resulta más fácil el tratamiento de números que el de letras.

nadores los emplea para distintas finalidades aunque, generalmente, contienen los gráficos y símbolos especiales. En el caso que nos ocupa, esta

El código ASCII incluye todos aquellos caracteres que pueden ser necesarios para cualquier comunicación.

zona está cubierta por la amplia gama de caracteres gráficos de que el ordenador dispone, una parte de ellos accesibles mediante **SHIFT** y la pulsación de la tecla correspondiente al gráfico, y el resto en combinación con la tecla **COMMODORE.**

Además, aunque esta aplicación nos es menos interesante, el BASIC utiliza estos códigos de carácter superiores al 127 para almacenar, dentro de la zona de texto BASIC, las palabras clave propias del lenguaje, consiguiendo así un importante ahorro de memoria y reducción en el tiempo de interpretación de las sentencias.

El estudio del código ASCII tiene un cierto interés, puesto que como ya hemos dicho, es empleado por una gran cantidad de ordenadores y periféricos muy importantes como, por ejemplo, la mayoría de las impresoras.

Fundamentalmente, podemos observar en él tres grandes bloques: el primero de ellos se extiende desde el 0 hasta el 31 y está constituído por caracteres de control. A este bloque siguen una serie de símbolos y signos de puntuación, que comienzan por el código 32 (espacio), y finalizan en el 64 conocido generalmente por arroba ( ). Dentro de este grupo es importante destacar la presencia de los dígitos, que se extienden desde el 48 (cero) hasta el 57 (nueve).

El siguiente bloque importante empieza en el código 65 y llega hasta el 90, conteniendo todo el alfabeto anglosajón en mayúsculas. Una pequeña serie de tan sólo seis caracteres cuyos códigos van del 91 al 96, separan este grupo del siguiente.

El último bloque a reseñar comprende los caracteres del 97 al 122, y representan de nuevo el alfabeto anglosajón, pero esta vez en minúsculas, seguido de los últimos caracteres especiales, los cuales abarcan desde el código 123 hasta el 127. Así mismo, y dependiendo del estado del teclado (pulsación de **SHIFT** y **COMMODORE**), estos códigos pueden corresponder a la representación de gráficos.

Debemos hacer, por último, la salvedad de que los códigos 192 a 223 son los mismos que los 96 a 127, así como los códigos 224 a 254 se corresponden con los 160 a 190; lo mismo que sucede con el 255 y el 126.

El hecho de que el ordenador utilice



El estudio del código ASCII tiene interés puesto que es el empleado por la mayoría de las impresoras.



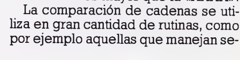
un código u otro puede ser que nos parezca en estos momentos carente de importancia, por tratarse de una cuestión de régimen interno que, en teoría, no afecta a la tarea de programación. No obstante, es éste un punto vital en el desarrollo de gran cantidad de programas.

Cuando el ordenador efectúa comparaciones numéricas, su lógica resulta aplastante: «4» es mayor que «1». Sin embargo, es sorprendente cómo realiza comparaciones entre letras y se muestra bien seguro de que la «B» es mayor que la «A»; y lo que es más, se obstina en que el símbolo «%» es mayor que el símbolo «$». Esto último ya no parece tan lógico. Aceptamos que suponga que la «B» es mayor que la «A», pero no sabemos la razón de que el símbolo de «%» sea considerado como superior al de «$».

El sistema que sigue el ordenador es bien claro, se basa en el orden que tienen los caracteres dentro del código ASCII. Así pues, cuando deseamos comparar dos caracteres cualquiera, el ordenador se limita a averiguar su posición relativa dentro del código ASCII, dando por mayor a aquél cuyo código resulta superior.

Si en vez de un solo carácter comparamos series de éstos, las complicaciones no aumentan en exceso. En este caso, se van comparando los caracteres de las series que ocupen posiciones homólogas, y en el momento en que uno de ellos resulte mayor que su correspondiente, la comparación se decantará hacia ese lado. Por último, en el caso de que una de las series sea más corta que la otra, y todos los caracteres hayan sido iguales, se considerará mayor la serie de caracteres más larga.

Así, por ejemplo, en la comparación de las cadenas «AHG» y «AHGA», se considera mayor la segunda expresión, puesto que los primeros, segundos y terceros caracteres son iguales entre sí pero, sin embargo, la segunda cadena es más larga que la primera. Si la comparación puede resolverse antes de llegar al final de cualquiera de las cadenas la longitud no se tiene en cuenta para nada; así, por ejemplo, la cadena «B» es mayor que la «AAAA».



La comparación de cadenas se utiliza en gran cantidad de rutinas, como por ejemplo aquellas que manejan se-

El hecho de que el ordenador utilice un código u otro puede ser punto vital en el desarrollo de gran cantidad de programas.





Las funciones **CHR$()** y **ASC()** relacionan al programador con el código que el ordenador emplea internamente

ries, depuran entradas de datos, etc... Así pues, para atender las necesidades de programación sobre un tema que se ha revelado de vital importancia, el BASIC dispone de dos funciones de conversión directa, las cuales relacionan al programador con el código que el ordenador emplea internamente. Estas dos funciones se denominan **CHR$()** y **ASC().**

LAS FUNCIONES CHR$() Y ASC()

La función **CHR$()** tiene como argumento un número comprendido entre 0 y 255, proporcionando una cadena de salida con el carácter cuyo código se especifica en el argumento. El siguiente programa obtiene el juego de caracteres completo del ordenador:

```
10 REM – JUEGO DE CARACTERES
20 PRINT," – CODIGOS ASCII –"
30 FOR I=30 TO 127:PRINT CHR$(I);:
   NEXT
```

Como experiencia, podemos probar a pulsar simultáneamente las teclas **COMMODORE** y **SHIFT** para apreciar cómo se altera el juego de caracteres representado en pantalla, en función del modo elegido: mayúsculas y gráficos, o mayúsculas y minúsculas.

Al margen de su utilidad como función de codificación del juego de caracteres, **CHR$()** tiene un gran interés en otro aspecto: la posibilidad de tratar información de cadena como si fuese numérica, lo cual nos deja el camino abierto a las estructuras de bucle.

Supongamos que deseamos representar en columna el alfabeto anglosajón (mayúsculas). Nosotros reduciremos mentalmente esta tarea a una operación muy sencilla: «para un carácter variando desde la A hasta la Z, escribamos cada vez el que corresponda».

Como ya sabemos, la traducción de este tipo de estructuras cíclicas al BASIC se lleva a cabo de una forma prácticamente directa, mediante un bucle **FOR NEXT.** Por ello, podremos escribir:

```
FOR I$=A TO Z:PRINT I$:NEXT
```

Sin embargo esta instrucción sería rechazada por el intérprete del BASIC, puesto que no podemos utilizar variables de cadena como índices de ciclo.

La solución a nuestro problema con-



Con **CHR$()** tenemos la posibilidad de tratar información de cadena como si fuese numérica.

siste en encontrar una relación numérica entre los caracteres a representar. Esta bien puede ser el código que les corresponde dentro del juego de caracteres, y por tanto, será la función **CHR$()** la que vendrá a sacarnos del apuro. La rutina anteriormente expuesta podría codificarse de la siguiente manera:

```
FOR I=65 TO 90:PRINT CHR$(I):NEXT
```

Esta instrucción puede traducirse al castellano diciendo: «para los códigos que van del 65 al 90 escribe el carácter cuyo código corresponde».

La función **ASC()** es inversa a **CHR():** A partir de un argumento de cadema produce el código numérico del carácter correspondiente. Más estrictamente diremos que obtiene el código del primer carácter de una cadena, devolviendo el valor 0 si la cadena está vacía. Así, obtendríamos el mismo resultado (65) en los dos supuestos siguientes:

```
PRINT ASC("A")
PRINT ASC("ABC")
```

También podemos comprobar que las funciones **CHR$()** y **ASC()** son inversas escribiendo lo siguiente:

```
PRINT A
PRINT CHR$(ASC("A"))
```

El hecho de que **CHR$()** opere en base a argumentos numéricos, no sólo reporta el beneficio de poder ser empleado en estructuras cíclicas, sustituyendo a los caracteres, sino que también le facilita el camino hacia las funciones definidas por el usuario. Efectivamente, hasta ahora hemos estudiado funciones que el lenguaje BASIC incorpora y proporcionan un determinado resultado; sin embargo, el BASIC dispone entre sus sentencias de una que aporta una gran flexibilidad en su programación, y lo que es más, una personalización de los programas: la posibilidad de que cada programador pueda definir sus propias funciones.

Aunque con ciertas restricciones respecto a la operatividad de las mismas, las funciones definidas contribuyen de una manera importante a la programación de calidad, aunque lamentablemente, el frecuente desconocimiento de su utilidad las aparta de muchos programas que podrían incluirlas, mejorando determinados aspectos.



La función **ASC()** es inversa a **CHR$()**.

DEFINICION DE FUNCIONES

La definición de funciones propiamente dicha se canaliza a través de la sentencia **DEF FN (DEFine FuNction)** seguida del nombre de la función y el parámetro entre paréntesis. De comienzo parece un poco lioso, pero pronto veremos que la realidad es otra. Estudiemos separadamente cada uno de los componentes expuestos.

Los nombres de función deben ajustarse a la norma general de construcción de los nombres de variables numéricas.

El parámetro debe ser necesariamente uno, no pudiendo ser ninguno o más de uno, y siempre encerrado entre paréntesis a continuación del nom-

La definición de funciones se canaliza a través de la sentencia **DEF FN.**





bre de la función. Este parámetro representa el argumento de la función definida y, por tanto, el resultado de ésta dependerá directamente del valor del mismo.

La forma en que ejecutamos una función definida es mediante la función basic **FN.**

En definitiva, **DEF FN** es una herramienta de gran utilidad con la cual debemos contar.

Vamos a definir una función muy simple. La llamaremos «P», y obtendrá la raíz cúbica de un número; así pues, la línea de definición será: **10 DEF FN P(X)=X ↑ 1/3,** donde **X** es el parámetro. Si nos damos cuenta, hemos antepuesto un número de línea a la definición. Esto se debe a que esta sentencia opera de forma similar a la **DATA,** y no puede ser empleada en el modo directo.

La forma en que ejecutamos una función definida es mediante la función BASIC **FN,** la cual tiene lógicamente una sintaxis muy similar a la definición: debe ir seguida del nombre de la función a ejecutar y del parámetro encerrado entre paréntesis. Cómo cualquier función BASIC, precisa de una palabra clave anterior que indique el destino del resultado. Por ejemplo, vamos a escribir el valor de **P,** creada anteriormente, para **X=8.** La instrucción será en este caso: **PRINT FN P(8).**

El resultado obtenido será 2, pero no es esto lo importante, sino que no hemos utilizado ninguna sentencia de asignación **(=, INPUT** o **READ)** para otorgar valor a **X.**

Esto no quiere decir que no se puedan utilizar variables dentro de la definición de una función sino que, cuando se ha utilizado un determinado nombre como parámetro, el BASIC no considera la existencia de ninguna variable con ese mismo nombre. De no ser así, es decir, en el caso de que una variable cuyo nombre no se emplea en los parámetros sí tome parte en la definición de la función, su valor se obtendrá del que tuviera dicha variable en el momento de la ejecución de **FN.** Veamos un ejemplo:

```
10 DEF FN P(X) = X*Y
20 Y = 3
30 PRINT FN P(2)
```

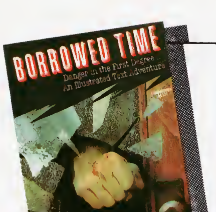
El resultado obtenido por esta función será 6 (2*3), puesto que el valor de **X** se tomará del parámetro y el de **Y** será extraído del área de variables. Por supuesto, no es necesario utilizar constantes al evaluar una función definida **(FN),** sino que podemos utilizar variables o expresiones. La única restricción es que figure exactamente el parámetro preciso; en caso contrario, obtendremos un error.

La utilidad de **DEF FN** es bastante grande ya que, además de ahorrar tiempo en la ejecución de los programas, nos permite definir algunas funciones interesantes presentes en otros BASIC's no COMMODORE. Así, por ejemplo, la función **MOD** da como resultado el resto de la división de dos números y puede ser simulada mediante: **DEF FN M(A)=A–B*INT (A/B);** asignando previamente a **B** el valor del segundo número.

Como último ejemplo, podemos definir la función de redondeo a cualquier número de cifras decimales, donde **X** es el número a redondear e **Y** el número de decimales requeridos: **DEF FN R(X) = INT ((X+(5/10 ↑ (Y+1)))*10 ↑ Y)/10 ↑ Y;** asignando, como en el caso anterior, el valor deseado a **Y** antes de ejecutar la función.

En definitiva, **DEF FN** es una herramienta de gran utilidad con la cual debemos contar.





Borrowed Time

Tú, el gran detective, duro de pelar como nadie, arrollador entre las mujeres, temido por tus enemigos, estás en estos momentos sentado en la mesa de tu despacho; quizás podríamos pensar que estás analizando duros casos criminales de personajes importantes que han acudido a ti como última solución a sus problemas, pero no es así. Como si dijéramos, la mesa de trabajo está totalmente patas arriba, tú te encuentras recostado en tu silla con tus piernas haciendo de puente entre ésta y la mesa, y pensando que el mundo se ha olvidado de ti, ya que nadie te ha llamado en dos semanas. Cuando de pronto...




l teléfana suena, tardas en reaccianar unas segundas parque tu cerebra na cree la que está escuchando, y acto seguido, abalanzándate sabre él, lo ases con fuerza mientras te la llevas a tu areja, pensanda en que será una incitante voz femenina pidienda pratección de un hérae cama tú. Pero cuanda las palabras camienzan a surgir par el hila telefónica, necesitas de unas mamentas para asimilarlas después de aír el «click» que indica que te han dejada «calgada». Piensas que te has abandanada a la inactividad y que tu cara can barba de tres días, tus manas sucias, tus lentas reflejas, son señales claras de que tu cuerpa está dejanda de funcianar, ¿y si es así, parque na puedes haber entendido mal a la misteriosa vaz?

Pero na, las palabras han llegada a ti con la claridad de un raya cegador, y han resonada en tu cabeza un buen rata, pravocándate unas imperceptibles palpitaciones en las sienes y una ligera subida del nivel cardiaca, sí, la vaz decía: «Sam, eres hambre muerta».

Descubre la terrible historia que se aculta tras estas palabras, lapidatarias, e intenta sobrevivir en el duro munda de Barrawed Time, aunque te advertimos que no te será muy fácil, muchas aspiran a volarte la cabeza, pero quizás na seas el que crees que eres y ésta no repasará entances mucho sabre tus hambros, pues estás en grave peligra, y a partir de que salgas de tu despacha te parecerá que el munda se ha vuelta en tu cantra, iánima!.. y que na escriban tu epitafio.



VEREDICTO FINAL

GRAFICOS	*****
SONIDO	*****
ORIGINALIDAD	****
DIFICULTAD	*****
INTERES	*****



FICHA TECNICA

NOMBRE	BORROWED TIME
PRECIO	3.700 PTAS.
SOPORTE	DISCO
TIPO	AVENTURAS
MODELO	C-64, C-128
OBSERVACIONES	NINGUNA

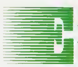


Golf Construction Set

El sol cae de pleno sobre The Belfry, el viento es del Noroeste, con una velocidad de 10 km por hora, los greens están más duros de lo que podíamos esperar, y nuestros lanzamientos se van consecuentemente, más lejos de lo que esperábamos. Nuestro caddy nos aconseja utilizar un hierro 5 para el próximo golpe; apuntando con precisión, nos aprestamos a realizarlo... el swing ha sido perfecto y la bola lleva buena trayectoria... pero ¡es increíble!, ¡ha encontrado! ¡es un «eagle»!

VEREDICTO FINAL

GRAFICOS	* * * * *
SONIDO	* * *
ORIGINALIDAD	* * * *
DIFICULTAD	* * * *
INTERES	* * * * *



El realismo de Golf Construction Set está muy conseguido, yo que podemos elegir el palo con el que queremos efectuar cada golpe (modero 1-5, hierro 1-9, PW, SW y putter), el punto donde queremos golpear la bola con el propósito de elevarla más o menos, el efecto o imprimirlo, la dirección... con lo cual no dejamos nada al ozar y podemos decir con seguridad que el golpe lo hemos efectuado sin duda nosotros.



El juego viene presentado en uno corcasa con dos cintas y cuotro caras distintas, de los cuoles una es para el programa principal, otra incluye el set de construcción de pistas, la siguiente lo escenarios donde se realiza la competición, y la última estó reservada para hacer tus propias grabaciones de hondicaps y hoyos. Si hemos decidido competir podemos elegir hacerlo a 18 hoyos (1-4 jugadores) o en modalidad Motch Play, es decir, a un sólo hoyo contra otro competidor, de tipo eliminativo. Uno vez has elegido el tipo de competición y número de jugodores, tienes la oportunidad de optor por un recorrido, entre los cuatro suministrados por el programa: The Belfry, Sunningdale, Wentworth Old Course, The Royal St. Georges, o uno que tú mismo quieras construir, en el cual puedes incluir las dificultades que desees, tales como órboles, logos, pendientes...

Asimismo, puedes imponerte tú mismo las condiciones en las que quieres jugar, tales como velocidad del viento, grados de temperotura, dureza del terreno... los cuales son muy importantes para el desorrollo del juego, pues influyen en dirección, penetrabilidod de la bolo en el aire, bote... Por último decir que una vez hayas acobado tu recorrido podrós grabar tu hondicap con respecto ol par del compo, paro empezar con él en otro ocasión, si no se te asignará siempre 28, que es el correspondiente a un principionte.



FICHA TECNICA

NOMBRE	GOLF CONSTRUCTION SET
PRECIO	2.500 PTAS.
SOPORTE	CINTA
TIPO	DEPORTES
MODELO	C-64, C-128
OBSERVACIONES	NINGUNA



Purple Turtles

Cuándo no nos ha ocurrido a cualquiera de nosotros de pequeños que ansiábamos algo con todas nuestras fuerzas y siempre algo nos lo impedía poseer; unas veces las prohibiciones provinientes de los «mayores», otras veces nuestras propias limitaciones físicas. La mayoría de las veces el objeto de nuestra ansiedad era únicamente una nimiedad, y los obstáculos que se nos habían interpuesto nunca hubieran bastado para impedirnos realizar nuestro propósito. Este es el caso de Purple Turtles.

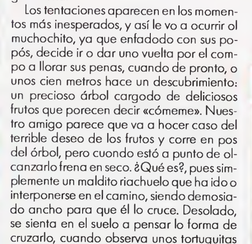
Cualquier coso, por pequeña e insignificonte que pueda parecer, atrae siempre la curiosidad de un niño, en nuestra historia nuestro protagonista es un goloso que gusto de estar o todas horos comiendo chucherías, pero su mayor interés recoe en la fruto: plátanos, manzonas, peros o cuolquier cosa parecida ejercen una atracción imposible de ig-



FICHA TECNICA

NOMBRE	PURPLE TURTLES
PRECIO	1.500 PTAS.
SOPORTE	CINTA
TIPO	HABILIDAD
MODELO	C-16
OBSERVACIONES	SIMPLICIDAD EXTREMA





norar para nuestro chiquitín, habiendo recibido en su cosa mós de uno bronca por tal glotonería.

Los tentaciones aparecen en los momentos más inesperados, y así le vo a ocurrir ol muchochito, ya que enfadodo con sus popós, decide ir o dar uno vuelta por el campo a llorar sus penas, cuando de pronto, o unos cien metros hace un descubrimiento: un precioso árbol cargodo de deliciosos frutos que porecen decir «cómeme». Nuestro amigo parece que va a hocer caso del terrible deseo de los frutos y corre en pos del órbol, pero cuondo estó a punto de olcanzarlo frena en seco. ¿Qué es?, pues simplemente un maldito riachuelo que ha ido o interponerse en el camino, siendo demosiado ancho para que él lo cruce. Desolado, se sienta en el suelo a pensar lo forma de cruzarlo, cuando observa unos tortuguitas (cuatro en totol) que con sus coporazones ocupon desde una orilla o otra. Sin arredrarse lo más mínimo, intenta cruzar subido en sus corazas, pero he oquí que se sube en una y ésta se sumerge, con el consiguiente chapuzón para el pequeño.

Aquí es donde entra nuestra inestimable oyudo, ya que deberemos guiar los saltos de este jovenzuelo para impedir que se moje, teniendo especial cuidodo en adivinar los movimientos de las tortugas, que se sumergirán en los momentos mós inesperodos. La dificultad en llegar hasto los frutos se irá incrementando a medida que voya avanzando el juego, así que mucho cuidadito y... a por la fruta.



VEREDICTO FINAL

GRAFICOS	* *
SONIDO	*
ORIGINALIDAD	* *
DIFICULTAD	*
INTERES	*



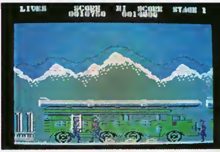
Green Beret

¿Queríais entrenamiento militar?, pues aquí lo tenéis, pero en este caso no es un entrenamiento, es la realidad. El general Jackson T. Kalliber nos ha enviado a una peligrosa misión, que consiste en rescatar a unos rehenes de los que ha conseguido apoderarse el enemigo, y se hallan en una verdadera pradera custodiada por multitud de artefactos militares y con una guarnición que supera en mucho a nuestras fuerzas de ataque, ya que en este caso constan de... una persona. ¿A que no adivináis quién es aquél que va a intentar la misión suicida de entrar en... Green Beret?



Ante todo debemos advertir el distinto peligro que entrañan cada hombre de los diferentes especialistas que cuenta la defensa enemiga. Los blancos, son denominados commanders, y son menos peligrosos que los demás, pero importantes, ya que si matamos a uno conseguiremos quitarle una de las armas especiales que porta; los verdes son maestros de kung-fu, los cuales van desarmados, pero claro, no les hacen falta armas; los marrones son las tropas regulares, no son especialmente peligrosos ni rápidos, su dificultad estriba en su número; y por último los azules, unos pájaros de mucho cuidado, ya que son los únicos que dispararán sobre ti, además sin previo aviso.

La defensa está dividida en cuatro niveles, cada uno con sus características especiales. En el primero habremos de sobrepasar unas plataformas de defensa muy conseguidas, lo cual será harto difícil si no conseguimos hacernos con un lanzallamas, para después pasar ante camiones lanzamisiles cuyos aledaños están sembrados de minas; el segundo nivel transcurre entre los barracones de las tropas enemigos y allí el peligro es enorme, puesto que nos sorprenderán con lanzadores de bazookas, muy certeros por cierto, e incluso con unidades de paracaidistas, mata al commander y róbale su bazooka, y llegamos al tercer nivel, donde pasaremos por alambradas repletos de enemigos y seremos atacados por terribles perros salvajes hasta conseguir acceder a la cuarta fase, donde las dificultades serán ya inimaginables, si logras acabar con esta pesadilla te convertirás en un verdadero héroe, si por el contrario fracasas serás uno más en la lista del cementerio más próximo a Green Beret.





VEREDICTO FINAL

GRAFICOS	*****
SONIDO	****
ORIGINALIDAD	***
DIFICULTAD	*****
INTERES	*****

FICHA TECNICA

NOMBRE	GREEN BERET
PRECIO	2.500 PTAS.
SOPORTE	CINTA
TIPO	GUERRA
MODELO	C-64, C-128
OBSERVACIONES	NINGUNA



Desert Fox

Entre las inmensas dunas del desierto, los soldados descansan sin saber qué es lo que les espera al despertarse, puede ser la muerte, o quizás consigan sobrevivir un día más en esta inmensa e inhóspita extensión de arena. Sus posibilidades de supervivencia son pocas, ya que en el bando contrario tienen nada menos que al Mariscal Rommel, el famoso «Zorro del Desierto», sobrenombre que se ha ganado a pulso, por su aureola de ser invencible en un terreno como éste. ¿Conseguirá alguien borrar este apodo?

Tus ataques contra las fuerzas de los Aliados en el norte de Africa comienzan a ser verdaderamente preocupantes, ya que con su astucia y sentido estratégico está empezando a abrir líneas en el frente. Por ese motivo, el Estado Mayor ha decidido tomar cartas en el asunto y jugársela al as de oros. Por unanimidad se ha decidido que el único hombre que podría detener ese avance es «Lobo Solitario», que como podréis adivinar somos nosotros. Por lo que somos enviados a





las traicioneras dunas a enfrentarnos a Rommel, con un objetivo concreto: rescatar las bases de combustible que nos han sido conquistadas.

Gozamos de algunas ventajas, ya que hemos logrado conectar con la onda de radio del mariscal, de forma que siempre que algún mensaje es enviado a otras bases, logramos interceptarlo para nosotros, sabiendo así de antemano los movimientos enemigos.

Tendrás otras misiones al tiempo, como repeler ataques de Stukas, a los que detectará el radar antes de que lleguen a tu posición; los ataques de los temibles tanques «Tigre», devastadores casi siempre, e incluso te tenderán emboscadas en un desfiladero, el cual deberás cruzar con los menores daños posibles en tus fuerzas. Otro de tus peligros estribará en que, en ocasiones, al trasladarte de una posición a otra, te toparás con campos plagados de minas, las cuales tendrás que evitar con la mayor diligencia posible, ¡el tiempo es oro!

Y ya por último deberás proteger los convoys aliados de los ataques aéreos del Zorro, si los destruyen perderás muchas posibilidades de ganar tu batalla, pero, ¡ánimo y a los cañones!



FICHA TECNICA

NOMBRE	DESERT FOX
PRECIO	2.600 PTAS.
SOPORTE	CINTA
TIPO	GUERRA
MODELO	C-64, C-128
OBSERVACIONES	BUEN SIMULADOR DE VOZ

VEREDICTO FINAL

GRAFICOS	****
SONIDO	*****
ORIGINALIDAD	****
DIFICULTAD	***
INTERES	****

¡Manos a la obra!

Como todos hemos oído alguna vez, a andar se aprende andando. Claro que en esta sección no os vamos a enseñar el pedestre movimiento (pedante, ¿verdad?), pero ha llegado el momento de enfrentarnos con el código máquina en la práctica.

ara poder realizar algo en código máquina, es necesario disponer de un ensamblador. En este ejemplo, y en los que seguirán, hemos utilizado el Machine Lightning, uno de los más completos que conocemos.

No obstante, nuestros ejemplos son de validez general, transportables de un ensamblador a otro. Incluso, para los que no dispongáis de uno, acompañamos los ejemplos de cortos listados en BASIC que os permitirán ejecutar los programas directamente.

Una última distinción: el Machine Lightning es un macroensamblador de dos pasadas. Esto significa que nos permite el uso de etiquetas, macros y operaciones aritmético-lógicas en el propio ensamblador, como ayuda en la programación.

El ensamblado (es decir, la traducción a código máquina) se hace en este caso en dos pasadas. En la primera, se busca el valor de las posibles etiquetas y se comprueban los errores. En la segunda, se produce el ensamblado propiamente dicho, con las etiquetas sustituídas por su valor.

Esto tiene la enorme ventaja de que no es necesario asignar el valor a las etiquetas «a mano» antes del ensamblado, ahorrando más de un quebradero de cabeza, y simplificando también el propio listado.

Y, una vez hechas las precisiones preliminares, manos a vuestros respectivos ensambladores... ¡Y al ataque!

Un primer ejemplo: SCROLL de una línea de la pantalla

Vamos a realizar una corta rutina que desplace los caracteres de la primera línea de la pantalla una posición a la izquierda; el carácter que desaparezca por el borde izquierdo aparecerá por el lado derecho (es decir, existirá un «wrap around» de la línea).

Antes de lanzar nuestros dedos sobre el teclado, pensemos. El proceso de scroll está esquematizado en el organigrama adjunto.

En primer lugar, señalaremos que esta rutina, tal como se presenta, se ha realizado pensando en el C-64. Sin embargo, los principios en ella expuestos son totalmente aplicables a los demás ordenadores Commodore, sin más que cambiar la dirección del inicio de la memoria pantalla y, en su caso, la longitud de la línea (para el Vic-20).

Como ya sabéis, los caracteres se hallan almacenados en una zona de memoria de 1000 bytes que comienza en la dirección $0400 (1024 decimal). Cada 40 bytes definen una línea de la pantalla.

En el comienzo del programa, el contenido de la posición $0400 (es decir, el carácter de la esquina superior izquierda de la pantalla) se almacena en la pila. Hacemos esto, pues de lo contrario se perdería, al ser sustituido por el carácter de su derecha, durante el scroll.

Seguidamente, iniciamos un bucle, desplazando uno por uno los 30 caracteres restantes de la línea:

Utilizando el registro X como puntero del carácter, y aprovechando el direccionamiento postindexado por X, tomamos el carácter de la posición siguiente y lo almacenamos en el actual; es decir, tomamos el contenido de 1024+1+(X) y lo depositamos 1024+(X).

Vemos aquí la necesidad de guardar previamente el contenido de la primera posición de la pantalla, pues éste será sustituído por el carácter inmediatamente a su derecha.

A continuación se incrementa el contenido del registro X mediante INX, y se compara con 39. Si es igual, el bucle se ha ejecutado ya 39 veces y salimos de él; si no, volvemos a iterar. Fijaos que hemos empezado con X=0 y terminamos con X=39.

Finalmente, recuperamos el byte almacenado en la pila y lo colocamos en la posición 1063, el carácter más a la derecha de la línea. Con esto, ya hemos creado el scroll.

Organigrama	Instrucción
INICIO	
TOMA (1024)	LDA $0400
GUARDA EN PILA	PHA
REGISTRO X=0	LDX # 0
TOMA (1025+X)	BUCLE LDA $0401,X
GUARDA (1024+X)	STA $0400,X
INCREMENTA X(X=1+X)	INX
(X) = 39 FIN DE LA FILA? (NO: vuelve a TOMA (1025+X); SI: continúa)	CPX # 39 BNE BUCLE
SACA DE PILA	PLA
GUARDA EN (1063)	STA $0427
VUELVE AL BASIC (RTS)	RTS
FIN	

Notad que en este caso no desplazamos los atributos de color de los caracteres, por lo que los colores permanecerán fijos. Este puede ser un buen ejercicio, a desarrollar por vosotros mismos.

Algunos detalles a considerar

Una vez que tenemos muy claros los fines, debemos poner los medios para lograrlos (teorema de Maquiavelo aplicado a la informática).

Un medio muy útil para clarificar las ideas (a veces, porque otras veces las oscurece, especialmente si lo hacemos mal...) es la realización de organigramas o diagramas de flujo. Como cualquier programador sabe, es el sistema más eficaz de ordenar el flujo de un programa.

```
==============================================================================
A LISTING, PAGE #1
==============================================================================


PASS 1


PASS 2

LINE# LOC. OBJECT LABELS          LINE

   10 C000                        *=$C000
   20 C000 AD0004                 LDA $0400
   30 C003 48                     PHA
   40 C004 A201                   LDX #$01
   50 C006 BD0004                 LDA $0400,X
   60 C009 9DFF03                 STA $03FF,X
   70 C00C E8                     INX
   80 C00D E028                   CPX #$28
   90 C00F D0F5                   BNE $C006
  100 C011 68                     PLA
  110 C012 8D2704                 STA $0427
  120 C015 60                     RTS

SUCCESSFUL ASSEMBLY; NO ERRDRS.
```

```
==============================================================================
A LISTING, PAGE #2
==============================================================================


   10               .DRG    49152      ;SITUACIDN PRDGRAMA
   20 INICIO       =1024               ;INICID MEMDRIA PANTALLA
   30               LDA INICIO         ;GUARDA EN STACK EL CDNTENIDD
   40               PHA                ;DEL PRIMER CARACTER DE LA LINEA
   50               LDX #1             ;1 AL REG. X.
   60 BUCLE         LDA INICIO,X       ;TOMA CARACTER
   70               STA INICIO-1,X     ;LO DEJA UNA PDSICIÓN A LA IZQUIERDA
   80               INX                ;INCREMENTA X
   90               CPX #40            ;COMPARA CON 40
  100               BNE BUCLE          ;SI NO IGUAL, VUELVE A BUCLE
  110               PLA                ;RECUPERA CARACTER
  120               STA INICIO+39      ;LO DEJA AL INICIO LINEA
  130               RTS                ;VUELTA DE LA RUTINA (RETORNA AL BASIC)
```

En lenguajes de alto nivel, como BASIC o Pascal, podemos permitirnos el lujo de prescindir de los organigramas en programas no excesivamente complicados, compensando el (más que probable) desorden con la potencia y la flexibilidad de las instrucciones a nuestra disposición.

En código máquina, sin embargo, debido a que la poca flexibilidad de las instrucciones disponibles nos obliga a una gran precisión en el desarrollo del software, un programa desordenado es, en un 99.999999% de los casos un programa que no funcionará, o que lo hará incorrectamente. Ya sabéis, el que avisa, no es traidor.

Aceptad pues un consejo: Si pretendéis realizar un programa mínimamente complejo: un organigrama puede ahorraros mucho tiempo y, sobre todo, muchos dolores de cabeza.

Podéis ver en los listados adjuntos la solución que hemos adoptado, prácticamente «traduciendo» al pie de la letra el organigrama.

Muy bien, pero... ¿y eso cómo se hace de verdad?

Para introducir el programa en nuestro Commodore, lo más práctico es disponer de un ensamblador.

En este caso, sólo hemos de copiar el listado 1. Ya sabéis, pueden existir ligeras diferencias con la sintaxis de vuestro ensamblador, especialmente en las pseudoinstrucciones. Un vistazo al manual solventará el problema.

El listado 2 contiene el mismo programa, pero escrito en un ensamblador más potente. Podéis observar que no se hacen saltos a direcciones, sino a etiquetas; al calcular los saltos el propio ensamblador nos ahorramos una buena dosis de trabajo.

Un detalle a indicar: podéis ver que las líneas del programa se hallan numeradas, del mismo modo que un programa BASIC.

Esto no es en absoluto una característica del código máquina, sino una cualidad del editor del ensamblador utilizado. En este caso, se emplean números de línea para ordenar las líneas del código fuente; en otros ensambladores se prescinde de ellas, y se utiliza un sistema similar al de un procesador de textos para generar el programa fuente.

Finalmente, en el listado 3 se incluye el programa en BASIC, que carga y ejecuta la rutina en código máquina. Para mezclar programas Basic y C/M, introducimos en líneas **DATA** los valores numéricos de todas las instrucciones, junto con un bucle **FOR–NEXT** que los lea e introduzca en la posición adecuada de la memoria.

Si no disponéis de ensamblador, lo único solución que os queda para desarrollar programas en código máquina es realizar el ensamblado o mano. Para ello, lo más indicado es escribir en un papel el listado, tal como lo introduciríomos en un ensamblador. Una vez terminado, y verificado en lo posible su correcto funcionamiento en teorío, pasamos al ensamblodo.

Con una tobla de mnemónicos al lodo, como la que oporeció publicoda en el número 1 de nuestra revista, vamos poniendo junto a codo instrucción el código numérico que le corresponda, es decir, número de instrucción y, si lo hoy, bytes bajo y alto de la dirección correspondiente, en este orden.

Cuondo lleguemos a un byte de direccionamiento relativo, lo mejor es dejar un espocio y terminor el ensamblado. Cuondo hayamos finolizado, podemos ir contando los valores de los soltos relativos, tal como vimos al hablar de ellos.

Uno vez llegodos a este punto, escribiremos en líneas **DATA** los valores, junto con los instrucciones adecuadas paro leerlos e introducirlos en memoria. Si hemos hecho bien este trabojo, indudablemente pesodo, nuestra rutino en C/M debe estar listo para funcionar.

Claro que no hoce falta decir que este proceso sólo puede empleorse para programas cortos. ¡Imaginad traducir a mano un programa de 40K de longitud!

Además, la labor de depuración («debugging») se complica extraordinariamente, sobre todo si, después de la traducción, nos domos cuenta de que se nos han olvidado tres instrucciones vitoles... Volver o empezor (y en esta ocosión no es para un óscar, sino para un harakiri, por lo menos).

En este caso, como no hemos introducido saltos a direcciones obsolutas de memoria, nuestra rutina es totalmente reubicable.

El efecto de scroll conseguido es un poco brusco. Existe la posibilidad de obtener un scroll fino, punto a punto. Pero ello requiere el empleo de interrupciones y un conocimiento en profundidad del chip de vídeo. Este será uno de los ejemplos del uso de interrupciones, tema que abordaremos próximamente.

```
10 J=.:FORI=49152TO49173:READA:J=J+A:POKEI,A:NEXT
                                                  -051-
20 DATA 173,0,4,72,162,1,189,0,4,157,255,3,232,224,40
,208                                              -033-
30 DATA 245,104,141,39,4,96                       -110-
40 IFJ<>2353THENPRINT"{CLR}ERROR EN DATAS!":STOP
                                                  -162-
50 REM ESTO ES UNA DEMOSTRACION:                  -098-
60 REM ***************************                -164-
70 PRINT"{CLR}[2 ESP]ESTO ES UN SCROLL EN CODIGO MAQU
INA!"                                             -165-
80 FORI=1TO120:SYS49152:FORJ=1TO100:NEXT:NEXT
                                                  -153-
```



Joystick Mouse/

Hoy, en nuestra sección de equipos, vamos a hablar de dos joysticks poco habituales en lo que se refiere a lo que normalmente entendemos por joystick.

En su afón por conquistar a los usuorios de este discutido periférico, los fabricantes ofrecen al posible comprador, toda una serie de ventajas para un uso más cómodo del ortilugio (autofuego, poro diestros y zurdos, etc.); en esta carrera, unos consiguen plenamente su propósito y otros sólo o medios, ounque siempre será el usuario el que determine qué tipo de joystick le viene mejor, ya que, aquí como en todo, sobre gustos no hoy nado escrito, y hay personos que manejan perfectomente un tipo de joystick que a otro, sólo les sirve para hacerse un nudo en los dedos.

Los dos periféricos que vamos a comentar hoy son, por un lado el JOYSTICK MOUSE de Rush Wore, y por otro el modelo QUICKSHOT IX de Spectrovideo.

Joystick Mouse

Paro denominor a este tipo de joysticks, tenemos otro nombre mós apropiado: RATON, y es precisamente de eso de lo que se trata.

En primer lugar, vamos a explicar brevemente lo que es un rotón. Un ratón, es un periférico de entrada que se utiliza frecuentemente en oplicociones del tipo CAD o Computer Aid Design (Diseño Asistido por Ordenador), y consiste en una esfera situada sobre uno carcosa que contiene fundamentalmente dos potenciómetros; uno se encarga de leer los desplozomientos sobre el eje X y el otro sobre el eje Y, dando así constontemente la información posicional del «rotón», esto información es debidomente enviada al ordenador, que se encarga de convertirla en lo posición de un cursor (de olta o baja resolución), sobre la pantalla. Paro el uso de este periférico, se necesita una superficie plono de ciertas di-







Joystick Esférico

mensianes (par ejempla la mesa), sabre la cual desplazaremas el «ratón»; al hacerla, el gira de la esfera sobre la mesa, determinará nuestra pasición sabre la pantalla, sienda su usa muy intuitiva.

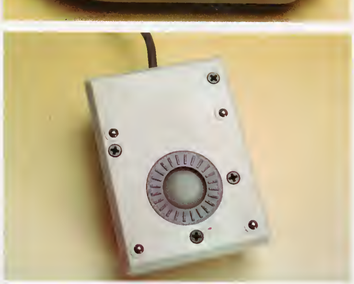
En el casa del Jaystick Mause de Rush Ware, este periférica viene camplementada can das batanes de «fuega» y una «interface» de canexión al PORT de jaystick de nuestra C-64; en la «interface» encantramas un interruptar can das pasicianes, de las cuales sóla funcianaró en el C-64 una de ellas. También encantramas das canectares tipa «jacks» hembras y un cable que las canecta; se trata de aditamentas para su funcianamienta en atras equipas (Atari, Schneider, etc.).

El equipa se acampaña de un pragrama de demastración denaminada «PAINT-BOX», que sirve para dibujar en mada multicalar. El programa en sí está muy canseguida par las múltiples apcianes que afrece; pera cama demastración del equipa deja mucha que desear, ya que es mucha más cómada y mós práctica el emplea de un jaystick canvencianal canectada directamente. De tadas madas, hay que decir que la pragramación de dispasitivas analógicas (cama la es el «ratón») canectadas a las parts del C-64, es bastante más campleja que la de dispasitivas digitales (jaysticks estóndar), par la que a la hara de trabajar can este pragrama, se aprecia que la sensibilidad del periférica dista mucha de ser la idónea, sienda, cama ya hemas camentada, recamendable el usa de un jaystick narmal.

En cuanta a las instruccianes del equipa, Dias nas libre de encantrar muchas así. En una palabra, FATAL; en la primera pógina nas encantramas can un par de dibujas que muestran ligeramente el mada de canexión al ardenadar, pera el texta (muy escueta), estó en alemán; unas póginas más adelante, nas encantramas can el misma texta pera esta vez en inglés, ademós, acerca de las interruptares de la «interface» na dicen absalutamente nada, y del cable que se acampaña, a duras penas se cansigue deducir que es para la canexión a Amstrad a Schneider, según parece verse en el primer dibuja; las instruccianes san para trabajar can el programa de demastración, y muestran el maneja de las funcianes de éste, que san:

DRAW: Dibujar.
RAYS: Dibuja rayas.









LINE: Dibuja una línea.
LINES: Dibuja líneas seguidas.
CIRCLE: Dibuja circunferencias.
DISK: Dibuja círculos o elipses rellenas.
FRAME: Dibuja rectángulos.
BOX: Dibuja rectángulos rellenos.
FILL: Rellena un área de dibujo con un color.
COPY: Copia trozos de dibujo en otra parte de la pantalla.
DOTS: Dibuja puntos.
SPRAY: Dibuja puntos aleatoriamente en una zona.
ZOOM: Activa el aumento de la imagen 8 veces.
COLOUR: Cambia de color una zona de dibujo.
DEVICE: Entra en la opción de cargar o grabar dibujos.
ERASE: Borra todo lo dibujado.
.: Anula lo último realizado.
TECLA RETURN: Intercambia pantalla de menú y de dibujo.

Joystick Esférico Quickshot IX

El aspecto que ofrece este joystick es impresionante, con sus dos botones de fuego rectangulares de 6.5x4.5 cm cada uno, su semiesfera de 10 cm de diámetro y su tamaño general de 21 x 14 cm, le hacen a uno sorprenderse la primera vez que la ve.

Pero; sin embargo, no debemos engañarnos, aunque a primera vista parece un «trackball», esto no es así, tratándose de un joystick convencional con un «mango» esférico (mejor dicho, semiesférico); mientras que en un «trackball» el control se consigue haciendo girar la esfera en todas direcciones, y por medio de potenciómetros se envía la señal analógica al ordenador. En este joystick, la señal es digital y la esfera no gira en todas direcciones de forma continua, sino que es el medio de cerrar los cuatro interruptores de dirección, como ocurre en los joysticks convencionales.

Joyball contra joystick o viceversa

Las características que tiene este joystick, aparte de su gran tamaño, son las siguientes:

Controlador esférico: Sustituye al típico mango convencional, y como ya comentamos al principio del artículo, será cada usuario el que decida si prefiere la esfera o el mango. En principio, parece más cómoda que el mango, ya que al controlar con suaves giros de muñeca, la fatiga disminuye bastante en uso prolongado (y a tope). En cuanto a resistencia, la batalla la gana el joyball al ser más insensible a los movimientos bruscos, pero pierde en cuanto a orientación, ya que el joystick parece más preciso en ese sentido.

Dos grandes botones de fuego: Aquí gana la baza el joyball, ya que te evita la necesidad de buscar ávidamente el botón de fuego en un apuro. Hay joysticks que tienen dos o más botones, pero los situados sobre el mango son poco recomendables debido a la pérdida de posición que puede ocasionar su pulsación; además, no son tan grandes como éstos que recuerdan un poco a los de los juegos de «máquina de bar». Ideales para los «arcade players».

LED indicadores de fuego: Encima de cada botón de fuego, se encuentran ubicados dos LEDs rojos que indican la pulsación del botón correspondiente; sólo sirven para saber si algún botón hace contacto permanente por avería, o para averiguar si estamos en fuego normal o autofuego (LED encendido continuamente o parpadeante), por lo demás, muy bonito pero poco útil.

Autofuego: Por medio de un interruptor muy bien disimulado en la carcasa, podemos seleccionar fuego normal o autofuego. Esta opción la tienen ya varios joysticks convencionales y es muy útil en los juegos de disparar a todo lo que se mueva. Por lo tanto: empate.

Ventosas de sujección: De nuevo es una característica común en joysticks normales, si bien en éste será más difícil «desventosarlo» de un tirón, debido a su controlador esférico en vez de por mango. Por consiguiente un nuevo empate.

Selector zurdo-diestro: Esta característica está menos vista en los joysticks, pero hay algunos (pocos) que la tienen; sirve para situar los botones de fuego al lado izquierdo o al lado derecho, sin invertir la lógica de direcciones. ¡Bien pensado!

Cable flexible de 5 pies: Esto es hoy día moneda corriente entre los joysticks, de todas formas es de agradecer, para poder situarnos a más distancia del monitor o TV, y para poder enrollarlo una vez hayamos terminado de usarlo. Nos encontramos ante otro empate.

En resumidas cuentas, es (como dice el fabricante) un cruce entre joysticks y trackball de excelentes resultados, todas las características son muy interesantes y es el usuario el que debe decidir (probándolo antes), si se adapta o no a este nueva periférica.







CHISPA

Con esta chispa, vamos a aprovechar la característica del C-64 para producir un scroll suave en los dos sentidos, haciendo variar los dos registros de forma aleatoria; con ello, conseguimos un efecto de vibración de la pantalla tipo terremoto, que puede sernos de utilidad, por ejemplo, cuando nuestro personaje se cae de la plataforma o cuando explota la bomba, o..., tú verás para qué.

```
10 FORI=0TO10
20 POKE53265,PEEK
   (53265)AND240ORRND
   (1)*8
30 POKE53270,PEEK
   (53270)AND240ORRND
   (1)*8
40 NEXTI:POKE53265,155:
   POKE53270,200
```

CHISPA

En el Basic del C-128 tenemos una infinidad de nuevas sentencias y comandos; algunos son verdaderas utilidades comparados con lo que habría que hacer en un C-64 para ejecutar la misma tarea, pero otros no son tan difíciles de implementar en el C-64, eso sí, con unas líneas de programa. La sentencia **INSTR** que se emplea así: **A= INSTR(A$,"LUNAR").** devuelve en la variable A la posición de la cadena "LUNAR" dentro de la variable A$. Esto es muy útil para controlar entradas en programas de tipo aventura o inteligentes. Para hacer exactamente lo mismo, utiliza el siguiente programa BASIC.

```
10 A$="EL PERRO TIENE
   LUNARES EN EL
   LOMO":B$="LUNAR"
20 FORI=1TOLEN(A$):
   FORJ=1TOLEN(B$)
30 IFB$=MID$(A$,I,LEN
   (B$))THENA=I
40 NEXT:NEXT:PRINTA
```

Aquí A$ contiene la cadena a examinar y B$ la subcadena a buscar, ejecutando **RUN** se imprimirá el valor (en la variable A) donde se encuentra la subcadena; si este valor es 0 es que la cadena A$ no contiene la subcadena B$.

CHISPA

Todos sabemos que la información para programar el sonido en el C-64 se debe colocar en ciertas posiciones de memoria, y que estas posiciones se repiten tres veces (una para cada voz). Empezando con el registro 0 de cada voz tenemos lo siguiente:

0: Byte bajo de frecuencia.
1: Byte alto de frecuencia.
2: Byte bajo de anchura de pulso.
3: Byte alto de anchura de pulso.
4: Forma de onda, modulación de anillo, sincronización, comienzo.
5: Ataque-decaimiento.
6: Sostenimiento-relajación.

Estos valores se repiten para cada una de las tres voces.

Bien, con el siguiente programa podemos actuar sobre ellos de forma fácil y sin tener que recordar todos los números:

```
10 DIM S(3,6)
20 FORI=0TO6
30 S(1,I)=54272+I
40 S(2,I)=54279+I
50 S(3,I)=54286+I
60 NEXT I
```

Ahora para situar cualquier valor, por ejemplo el ataque-decaimiento de la voz 2, actuaremos así: **POKE(2,5),** valor. Indicando el 2 el número de voz y el 5 el tipo de registro al que accedemos.

CHISPA

Si quieres controlar el tiempo de ejecución de un determinado fragmento de programa, puedes usar el siguiente truco:

```
10 TI$="000000":REM
   PARA PONER TIEMPO A
   CERO
20 FORI=0TO9999:NEXT:
   REM TROZO DE
   PROGRAMA A
   CRONOMETRAR
30 T=TI:PRINTT/60
   "SEGUNDOS":REM
   CALCULO DEL TIEMPO
   EN SEGUNDOS
```

Si necesitas afinar más, puedes imprimir directamente la variable T, pero debes de tener en cuenta que el valor que devuelve son sesentavos de segundo.

CHISPA:

A primera vista parece que el Datassette es un ente aislado que no tiene nada que ver con el ordenador; nada más lejos de la realidad, ya que como estamos acostumbrados a ver, el Datassette se para y se pone en marcha solo, cuando el C-64 lo necesita. Para poder controlar el motor por nuestra cuenta, aquí tenemos una serie de líneas de gran utilidad:

Estando el motor del Datassette en marcha (PLAY, RECORD & PLAY, REWIND o FFWD), esta línea lo detiene: **POKE192,1:POKE1,PEEK (1)OR32.**

Estando el motor parado pero alguna tecla de movimiento pulsada, esta línea lo pone en marcha: **POKE1, PEEK(1)AND39:POKE192,0.**

Si desde un programa queremos esperar a que se pulse la tecla STOP, estando pulsada alguna tecla de movimiento: **WAIT1,16.**

Si desde programa queremos esperar a que se pulse PLAY u otra tecla de movimiento: línea **IF PEEK(1)=55 THEN línea**.

CHISPA

Para los usuarios de la unidad de disco 1541 con el C-64, es desesperante el parpadeo de la luz roja como mensaje de error, teniendo que construirse una pequeña rutina BASIC para leerlo. Con esta chispa se carga una rutina de código máquina en la posición 828 (buffer del cassette), que con **SYS**828 nos revelará el motivo del parpadeo de la lucecita (error). Aquí está el cargador BASIC:

```
10 DATA 169,8,32,180,255,
   169,111,32,150,255,32,
   165,
20 DATA 255,32,210,255,
   201,13,208,246,32,171,
   255,96
30 FORI=828TO851:READ
   A:POKEI,A:NEXTI:
   SYS828
```

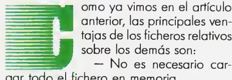
CHISPA:

El equivalente del comando BASIC **NEW** es en código máquina **SYS42564.** Esta rutina ROM puede ser utilizada cuando necesitamos el comando desde algún programa en código máquina.

Para aquellos propietarios del Commodore 128 que tengan curiosidad por saber el nombre de las personas que han diseñado su ordenador, pueden verlo reflejado en la pantalla tecleando **SYS32800, 123, 45, 6.**

Un, dos, tres, relativos otra vez

Debido a la gran aceptación que tuvo el artículo ¿RELATIVOS?, ¡SI GRACIAS!, y ante la demanda de una explicación más profunda sobre este tipo de ficheros en disco, vamos a dedicar este número de SOFTWARE, de nuevo al mismo tema.



Como ya vimos en el artículo anterior, las principales ventajas de los ficheros relativos sobre los demás son:

— No es necesario cargar todo el fichero en memoria.

— Necesidad de estructurar el fichero.

— Acceso directo al dato elegido.

— Gran velocidad de acceso, y por lo tanto de trabajo.

— Manejo de ficheros más grandes que la memoria del ordenador.

Ahora, vamos a ir viendo paso a paso como trabajar con estos ficheros, aclarando los puntos más oscuros según vayan surgiendo. Para ello hemos confeccionado un programa-ejemplo cuyo cometido es manejar un fichero de 365 registros (uno para cada día del año), los cuales están divididos en cinco campos que son:

Temperatura máxima:	2 bytes
Temperatura mínima:	2 bytes
Presión atmosférica:	3 bytes
Humedad relativa:	2 bytes
Comentario diario:	30 bytes

Como vemos, lo primero que debemos hacer para trabajar con relativos, es definir las necesidades de almacenamiento, esta fase es muy importante, ya que no se puede variar el tamaño de un registro una vez creado, si no es creando un nuevo fichero. Una vez conocido el tamaño de cada campo, procedemos a calcular el tamaño del registro; para ello, basta con sumar la longitud de todos los campos Y SUMAR UN BYTE POR CADA CAMPO. La razón de esto es facilitar la inserción de un separador entre campos que puede muy bien ser una coma. Por tanto, el cálculo será: 2+2+3+2+30+5=44 bytes (el último 5 es debido a que hay cinco campos por registro).

Para facilitar las explicaciones del texto, vamos a ir siguiendo las líneas de los tres programas ejemplo que se acompañan; el primero de ellos lo único que hace es crear el fichero; el segundo escribe datos de temperatura máxima, mínima, presión y humedad, de forma aleatoria y un comentario también aleatorio (que no tiene porque ser coherente) elegido de entre once posibles, este programa no es necesario si quieres utilizar el fichero en serio, aunque tampoco estorbará su uso, ya que el tercer programa (que es el bueno) se encargará de leer o escribir en cualquier registro del fichero.

Pues bien, vamos con el primer programa. La línea 100 es la encargada de abrir los canales de errores y de creación, como vemos, al final hay un **CHR$(44)** que es el que indica la longitud de cada uno de nuestros registros (calculados arriba); debemos fijarnos en los parámetros de la segunda sentencia **OPEN:** en primer lugar aparece un 2 que es el número de nuestro fichero, después un 8 del periférico y a continuación un 3 que es el número de canal, el cual es importante, como veremos más adelante. A continuación se sitúa una COMA, que no debemos olvidar para evitar disgustos; seguidamente el nombre del fichero precedido por el número de la uni-



dad "Ø:" o "1:" si tenemos dos, luego aparece otra COMA una L y otra COMA antes de cerrar comillas, terminando con **+CHR$** (longitud del registro).

En la línea 110 vamos a ver el comando de POSICIONAMIENTO; una de las cualidades de este tipo de ficheros, es que se pueden ampliar simplemente escribiendo algo en el registro deseado, por ejemplo, nosotros vamos a crear un fichero de 365 registros, para ampliarlo a 400 bastaría con escribir algo en el registro número 400. Análogamente, lo primero que hay que hacer es escribir cualquier información en el último registro que planeemos utilizar, de esta forma, obligamos al DOS (Sistema Operativo de DISCO) a CREAR todos los registros intermedios, si no sabemos la longitud que alcanzará nuestro fichero, escribamos algo en algún registro final aproximado. Pero vayamos con la sentencia en sí; para mandar el comando "P" se emplea el número de fichero con el que hemos abierto el CANAL DE COMUNICACIONES (15), que deberá ser el primero en abrirse y el último en cerrarse; en nuestro caso es el número 1, después aparece una COMA y entre comillas el comando "P" (posicionamiento) seguido de 4 sentencias **CHR$;** la primera de ellas lleva el número de CANAL con el que hemos abierto nuestro fichero (el número 3 que comentábamos anteriormente) + la CONSTANTE 96 (esto es "dogma de fe", se pone y punto). Los dos siguientes **CHR$** son los que determinan el número de registro en el formato byte bajo-byte alto; para calcular esto se usa la siguiente fórmula:

byte alto = INT (número/256)
byte bajo = número-byte alto* 256

Calculado en este orden para 365 tenemos: byte bajo = 109 y byte alto = 1, que no son otra cosa que los números que aparecen en el segundo y tercero **CHR$.** El cuarto **CHR$** se refiere al byte dentro de ese registro, por ejemplo, podríamos posicionarnos para sacar solamente el quinto campo (nuestro comentario) mandando un **CHR$(13),** no obstante, es recomendable leer o escribir registros enteros para no perdernos; además, queramos o no, el DOS lee información de 256 en 256 bytes, con lo que para leer aunque sólo fuese una letra, éste leería 255 más. Por lo tanto, nosotros emplearemos siempre un 1 (primer byte del registro).



En la línea 120 escribimos ya cualquier cosa en el registro 365. Obsérvese que el comentario no tiene 30 caracteres; no importa, este 30 es el número MAXIMO. Por esta razón, los demás campos pueden tener menos caracteres que su máximo dando problemas a la hora de extraer un determinado campo en vez del registro completo.

Por último la línea 130 cierra nuestro fichero, inicializa (NO VALIDA, que inutilizaría todos los ficheros relativos del disco) la unidad y cierra el canal de comunicaciones.

Hay que añadir que no se pueden tener dos o más ficheros relativos abiertos A LA VEZ, pero sí uno más del tipo normal (comunicaciones, secuencias, user). También es importante decir que con este tipo de ficheros, la opción@: (grabar y reemplazar) no funciona, con lo que los ficheros se deben leer o escribir pero nada más.

El segundo programa, es el encargado de rellenar el fichero con valores aleatorios, si no quieres utilizarlo no lo hagas, pero para ver el funcionamiento del tercer programa, tendrás que escribirte tus propios datos; si decides emplearlo, no pasará nada ya que podrás solapar lo que haya escrito con tus propios datos.

En la línea 100 hay que notar dos cosas, la primera es que la forma de abrir un fichero relativo ya creado es lo más simple que podamos imaginar **OPEN2, 8, 3,...;** sin necesidad de decir si es relativo ni si vamos a escribir o a leer, pues el DOS se encarga de todo. La segunda es la necesidad de tener una rutina que se encargue del control constante del canal de errores **(GOSUB 370).**

Antes de continuar, aclararemos que aunque los datos sean numéricos, es preferible trabajar siempre con CADENAS (string), de lo cual se encargan las segundas partes de las líneas 160-190 y la 210. En la 110 definimos nuestro carácter separador (S$) que no es más que una COMA (**CHR$(44)).** En la línea 140 se calculan los bytes alto-bajo del número de registro (K) y en la 150 nos posicionamos en el primer byte de ese registro; calculados unos valores aleatorios para cada campo, la línea 210 se encarga de formar la supercadena R$ que es la que se escribirá en el registro deseado en la 220. A continuación simplemente cerramos los ficheros y ya tenemos nuestro fichero ambiente "lleno".

A continuación, vamos a ver el tercer y último programa, que es el encargado de leer y escribir POR REGISTROS.

Desde la línea 100 hasta la 150 se encarga de escribir el mini-menú y de aceptar una entrada válida.

160-190 Es la rutina de lectura e impresión de resultados. Obsérvese la línea 170 de control de entradas, siempre que se pueda, DEBEREMOS controlar que las entradas sean válidas, para no enredarnos en errores incomprensibles, si no podemos hacer esto, habrá que ir constantemente a la rutina de control de errores del DOS (320-350).

200-210 Cierra los canales y termina el programa.

220-230 Rutina de impresión. En 230 se calculan byte alto y bajo del número de registro. Obsérvese que en la 250 leemos con 5 variables, mientras que escribíamos con una gran cadena (R$) en el segundo programa.

310-350 Lectura de canal de errores. En la línea 340 se comprueba que el registro sea válido (error número 50).

360-370 Abre los canales.

380-410 Espera una tecla.

420-580 Rutina de escritura, aquí se controla de nuevo la validez de los datos para que no excedan de las longitudes máximas (líneas 440, 520 y 540). En la 450 se define el carácter separador (**S$=CHR$(44)**, y en código que corresponde al carácter coma y nada tiene que ver con la longitud del registro, que por casualidad es también 44 en nuestro ejemplo) y en la 550 se forma la cadena total R$ que se escribe en 560.

Recomendaciones finales

Comprueba el canal de errores cada vez que se haga una operación.

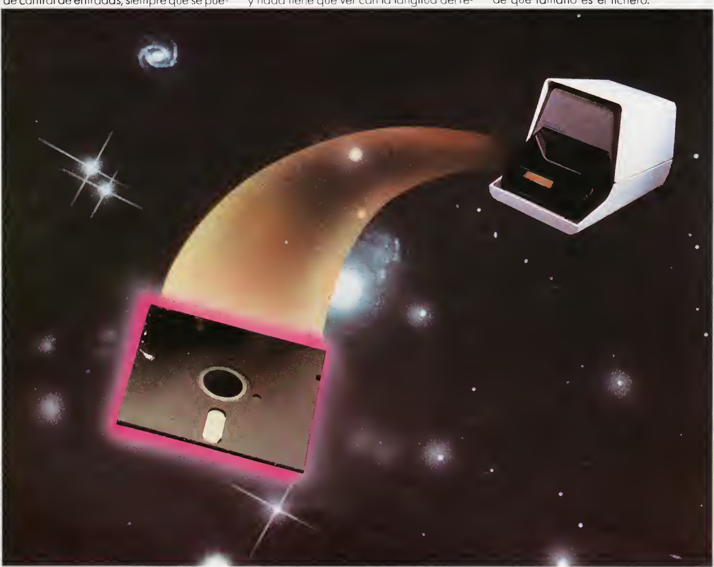
Antes de leer o escribir, habrá que enviar SIEMPRE un comando de posicionamiento.

Comprueba con lupa la sintaxis de TODOS los comandos del disco (apertura, posición, cierre, errores,... etc.).

El comando de posición se escribe en el número de fichero del DOS (canal 15).

Después de la «P» se envía un **CHR$** con el número de canal empleado al abrir NUESTRO fichero + 96.

En ficheros de longitud desconocida o variable "desperdicia" el primer registro para escribir el número máximo de registros alcanzado. Así, lo primero que hace el programa, es leer ese primer registro y saber de qué tamaño es el fichero.



LISTADO

LISTADO 1

```
100 OPEN1,8,15.OPEN2,8,3,"0 AMBIENTE,L,"+CHR$(44)
                                                  -019-
110 PRINT#1,"P"CHR$(3+96)CHR$(109)CHR$(1)CHR$(1)
                                                  -100-
120 PRINT#2,"11,10,700,55,FINAL"                  -227-
130 CLOSE2:PRINT#1,"I":CLOSE1                     -017-
```

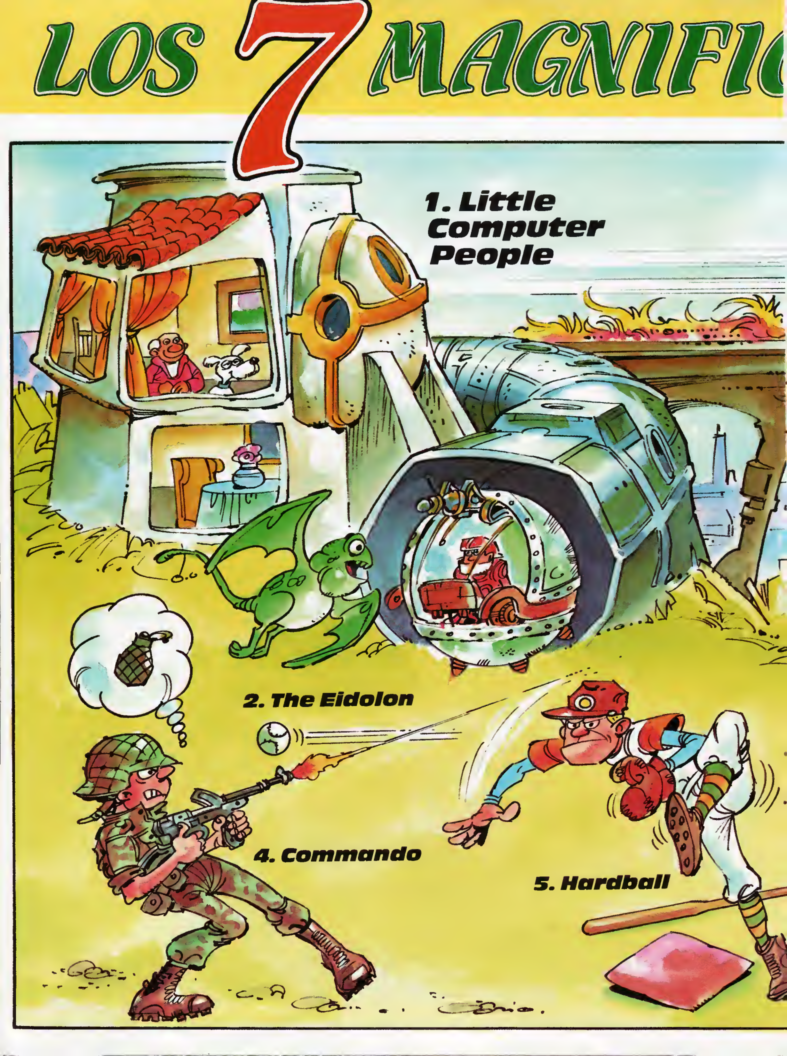
LISTADO 2

```
100 OPEN1,8,15.OPEN2,8,3,"0:AMBIENTE"·GOSUB370
                                                  -222-
110 S$=CHR$(44)                                    032-
120 DIMA$(10):FORI=0TO10:READA$(I) NEXT           -077-
130 FORK=1TO365                                    149-
140 AK=INT(K/256):BK=K-AK*256                     -160-
150 PRINT#1,"P"CHR$(3+96)CHR$(BK)CHR$(AK)CHR$(1)
                                                   182
160 MA=INT(RND(1)*30):MA$=MID$(STR$(MA),2)        -131-
170 MI=MA-INT(RND(1)*10).MI$=MID$(STR$(MI),2)
                                                  -095-
180 PR=INT(RND(1)*30)+690:PR$=MID$(STR$(PR),2)
                                                  -139-
190 HU=INT(RND(1)*70)+10·HU$=MID$(STR$(HU),2)
                                                  -067-
200 I=INT(RND(1)*11)                               102-
210 R$=MA$+S$+MI$+S$+PR$+S$+HU$+S$+A$(I)           140
220 PRINT#2,R$                                    -056-
230 NEXTK                                         -063-
240 CLOSE2.CLOSE1                                  063-
250 END                                           -142-
260 DATA HACE MUCHO FRIO                          -239-
270 DATA HACE MUCHO CALOR                         -049-
280 DATA ESTA NUBLADO                              070-
290 DATA NIEBLA MATINAL Y HELADA                  -254-
300 DATA CIELO TOTALMENTE DESPEJADO                037-
310 DATA CALINA TORMENTOSA                        -193-
320 DATA ANTICICLON PARA RATO                      109-
330 DATA NEVADA IMPRESIONANTE                     -157
340 DATA LLUVIOSO TODO EL DIA                      099-
350 DATA ... Y SIGUE LLOVIENDO                    -094-
360 DATA FRIO TERRIBLE                             156-
370 INPUT#1,A,B$,C,D.IFA<20THENRETURN              051-
380 IFA<>50THENPRINTA;B$,C;D:STOP:RETURN          -144-
```

LISTADO 3

```
100 POKE53280,0:POKE53281,0 PRINT"{CLR}"TAB(205)"{RON
}{GRN}1{ROF} LEER"                                 226-
110 PRINTTAB(45)"{RON}{YEL}2{ROF} ESCRIBIR"        -251-
120 PRINTTAB(45)"{RON}{CYN}3{ROF} TERMINAR"         013-
130 GETA$ IFA$=""THEN130                           -107-
140 A=VAL(A$).IFA>3ORA<1THEN100                    -086-
150 ONAGOTO160,430,200                             -229-
160 INPUT"{CLR}{GRN}PRIMER REG., ULTIMO REG.",R1,R2
                                                   -247-
170 IFR1<1ORR1>365ORR2<1ORR2>365THEN160            -144-
180 GOSUB370                                       -211-
190 FORI=R1TOR2STEPSGN(R2-R1).GOSUB230 GOSUB390 NEXT:
CLOSE2.CLOSE1:GOTO100                              -095
200 CLOSE2.CLOSE1                                  -059-
210 END                                            -138-
220 :           REM IMPRESION              :
                                                   -034-
230 AI=INT(I/256).BI=I-AI*256                      -150-
240 PRINT#1,"P"CHR$(3+96)CHR$(BI)CHR$(AI)CHR$(1)
                                                   -178-
250 INPUT#2,MA$,MI$,PR$,HU$,CO$                    -033-
260 PRINT"!U!!38 C!!I!",                           -084-
270 PRINT"!B![10 ESP]{RON}REGISTRO NUMERO"I;"{ROF}"SP
C((I>99)+(I>9)+10)"!B!";                           -231-
280 PRINT"!B!MAX: "MA$" MIN: "MI$" PRE: "PR$" HUM: "H
U$:PRINT"{IZQ}!2 B!NOTA  "CO$                      -221-
290 PRINT"{IZQ}!B!!J!!38 C!!K!{ABJ}"               -131-
300 RETURN                                         -147-
310 :           REM LECTURA ERROR          :
                                                   -166-
320 INPUT#1,A,B$,C,D·IFA<20THENRETURN              -046-
330 IFA<>50THENPRINTA;B$,C;D STOP:RETURN           -139-
340 IFA=50THENPRINT"REGISTRO NO VALIDO" STOP
                                                   -180-
350 RETURN                                         -152-
360 :           REM ABRIR CANALES          :
                                                   -120-
370 OPEN1,8,15:OPEN2,8,3,"0·AMBIENTE":GOSUB320·RETURN
                                                   -252-
380             REM ESPERA TECLA           .
                                                   -092-
390 PRINTTAB(12)"PULSA UNA TECLA"                  -042-
400 GETT$ IFT$=""THEN400                           -145-
410 RETURN                                         -149-
420 :           REM ESCRIBIR               :
                                                   -225-
430 INPUT"{CLR}{YEL}PRIMER REG., ULTIMO REG.";R1,R2
                                                   -119-
440 IFR1<1ORR1>365ORR2<1ORR2>365THEN430            -144-
450 S$=CHR$(44):GOSUB370                           -123-
460 FORI=R1TOR2STEPSGN(R2-R1)                      -122-
470 AI=INT(I/256):BI=I-AI*256                      -156-
480 PRINT#1,"P"CHR$(3+96)CHR$(BI)CHR$(AI)CHR$(1)
                                                   -184-
490 PRINT"{ABJ}[11 ESP]{RON}REGISTRO NUMERO"I
                                                   -191-
500 PRINT "{ABJ}MAXIMA,MINIMA,PRESION Y HUMEDAD"
                                                   -056-
510 INPUT "MA,MI,PRE,HU";MA$,MI$,PR$,HU$           -136-
520 IFLEN(MA$)>2ORLEN(MI$)>2ORLEN(PR$)>3ORLEN(HU$)>2T
HEN510                                              098-
530 PRINT "{ABJ}NOTA DEL DIA (MAXIMO 30 CARACTERES)"
                                                   -139-
540 INPUT"COMENTARIO";CO$ IFLEN(CO$)>30THEN540
                                                   -135-
550 R$=MA$+S$+MI$+S$+PR$+S$+HU$+S$+CO$             -074-
560 PRINT#2,R$                                     -063-
570 PRINT                                          -073-
580 NEXTI:CLOSE1:CLOSE2.GOTO100                    -012-

READY.                                             -173-
```


LOS 7 MAGNIFIC
1. Little Computer People
2. The Eidolon
4. Commando
5. Hardball



Colaboremos todos en la confección de estos 7 MAGNIFICOS de TU MICRO COMMODORE. Envianos el nombre de tu programa favorito, dejando muy claro tu nombre y dirección. Todos los meses, sortearemos cinco suscripciones por un año a nuestra revista entre las respuestas recibidas. Anímate y escríbenos a: TU MICRO COMMODORE. (7 MAGNIFICOS). Apartado de Correos 61.294. 28080 MADRID.

7. The Way of the Tiger

3. Back to the Future

6. Ballblazer

The Music Studio

«THE MUSIC STUDIO» es una herramienta musical de calidad profesional, encaminada a la creación de sonidos y canciones. Este programa pone la potencia del proceso de datos por ordenador al servicio de la composición, edición, orquestación y grabación instrumental; siendo, además, de muy sencillo manejo, tanto para compositores noveles como profesionales, a la vez que está dotado de gran potencia y flexibilidad.



En definitiva, podemos decir que se trata, al menos por el momento, del programa más completo desarrollado para la composición musical en los ordenadores: COMMODORE AMIGA, COMMODORE 128, COMMODORE 64 y ATARI ST; diseñado y desarrollado por Audio Light, Inc. para Activision y distribuido en España por Proeimsa.

Características fundamentales

El programa permite crear, modificar y almacenar composiciones musicales, empleando para ello una gran variedad de instrumentos y efectos sonoros, estando permitida la conexión vía interface tipo **MIDI** a instrumentos musicales electrónicos de alta calidad; convirtiendo el conjunto en un potente sintetizador musical.

Una vez concluída la composición, es posible volverla a escuchar, empleando los instrumentos seleccionados de entre una gran variedad, almacenados en una «biblioteca instrumental», visualizándose al tiempo en pantalla la partitura musical generada, con todas sus peculiaridades.

Estas son algunas de sus características más destacadas:

—Fácil manejo a través del «ratón».

—Rapidez en la selección de opciones, por el posicionamiento y pulsación del botón.

—Identificación de las opciones y simbología en uso en la parte superior de la pantalla.

—Posibilidad de empleo conjunto con un instrumento musical electrónico MIDI-compatible, para obtener una calidad de sonido profesional.

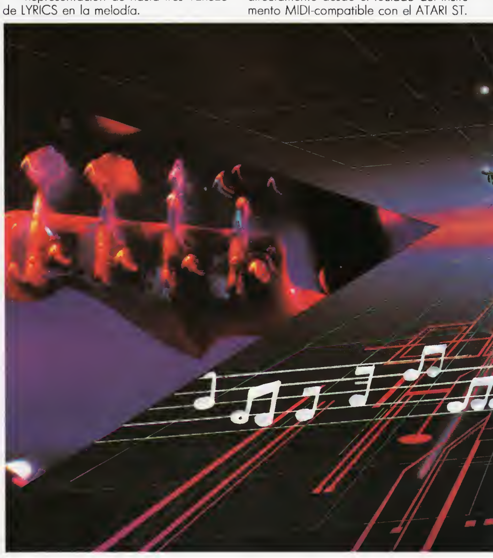
—Adopción de la notación musical convencional, incluyendo SHARPS, FLATS, RESTS, TIES, DOTTED NOTES y TRIPLETS.

—Intercambio automático, en un solo paso, de funciones de teclas.

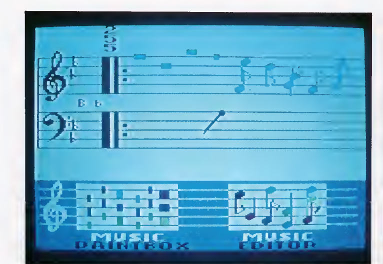
—Representación de hasta tres VERSES de LYRICS en la melodía.

—Posibilidad de definir instrumentos musicales inéditos, potenciando los parámetros del instrumento MIDI-compatible.

—Posibilidad de introducción de notas directamente desde el teclado del instrumento MIDI-compatible con el ATARI ST.







C-64



C-64

Las cinco pantallas de trabajo

Podemos considerar como pantalla principal del programa a la que permite la composición y edición. Dentro de esta pantalla pueden distinguirse tres zonas. Tanto en la superior como en la inferior se encuentran los símbolos que, por selección directa, permiten ejecutar diferentes trabajos; mientras que, en la franja central, se desarrolla la representación musical propiamente dicha, sobre el pentagrama.



En la pantalla de diseño de instrumentos musicales y efectos de sonido, podemos trabajar a partir de las cualidades básicas predefinidas para cada instrumento convencional, dotándolos de efectos sonoros especiales, o definir uno inédito; moviéndonos siempre para ello dentro de las capacidades sonoras de cada ordenador.

La pantalla de parámetros para el MIDI, permite la asignación de nuevos instrumentos y efectos sonoros, sirviendo además para determinar el rango de octavas del instrumento externo.

La pantalla de «caja de pinturas» musical, permite la composición sin el empleo de la notación musical stóndar; contando para ello con una gama de colores que identifican las notas y diferentes anchos de barras para determinar sus duraciones.

Por último, la pantalla de Menú de Ficheros permite la grabación y carga de composiciones musicales o efectos sonoros, tanto de los creados por el usuario, como de los ya existentes en el disco del programa, a modo de demostración de las posibilidades musicales del mismo.

La zona de mensajes

Una de las características que hace más sencilla la utilización del programa es la zona de mensajes, situada en la parte superior de la pantalla principal de composi-



C-64

ción. Esta zona nos permite conocer el significado de cada símbolo, al apuntarlo con el ratón, y qué acción se encuentra en proceso.

Carga de una canción

«THE MUSIC STUDIO» viene dotado de composiciones pre-programadas almacenadas en el disco de programas. Estas composiciones pueden servirnos de ejemplo para conocer las posibilidades del programa, al tiempo que para facilitarnos el aprendizaje del manejo del mismo.

Para cargar en memoria una canción desde el disco del programa basta con seleccionar el Menú de Ficheros, situado en la parte superior izquierda de la pantalla y, manteniendo el botón del ratón presionado, seleccionar SONG FILES. Con ello, tendremos acceso a una lista de títulos que aparecerán al margen izquierdo de la pantalla.

A continuación, debemos posicionar el ratón sobre la palabra LOAD y seleccionar nuevamente con el ratón cualquiera de los títulos que aparecen en la pantalla, con lo que el nombre de la canción seleccionada aparecerá en la zona de mensajes al tiempo que se produce la carga; con un mensaje de espera, el cual precede al de FILE LOADED, indicativo de haberse completado la operación.

Basta ahora con desplazar el ratón hasta la palabra DONE y pulsar un botón, con lo que el programa se desplaza automáticamente hasta la pantalla principal de composición, sobre cuyo pentagrama aparecerán las primeras notas del título seleccionado anteriormente.

Reproducción de una canción

Puede escucharse el título seleccionado con las notas desplazándose horizontalmente a través de la pantalla, o sin desplazamiento. Para determinar cada opción basta con situar el ratón sobre el símbolo de la nota en movimiento o en el de la oreja, respectivamente. Con ello, comenzará a sonar la canción.

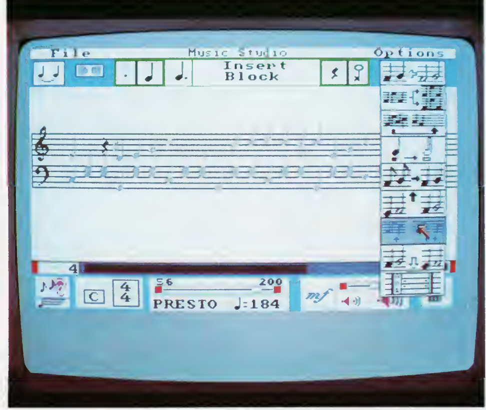
Al tiempo que se van escuchando las notas, en la parte inferior de la pantalla, va desplazándose un rectángulo a través de una barra, para indicar la posición relativa de la nota en curso frente al total de las que se compone la canción. Al mismo tiempo, aparece a la izquierda de esta barra un contador indicando el número de la columna de notas que se está interpretando y, en el extremo derecho, el número total de columnas de que se compone la canción.

En cualquier momento se puede recomenzar la interpretación, apuntando con el ratón al número situado a la izquierda de la barra horizontal. Del mismo modo, puede saltarse al final desplazando el ratón al número situado en el extremo derecho de la barra, con intención de añadir nuevas notas a la composición.

Además, puede seleccionarse la interpretación de la melodía columna a colum-



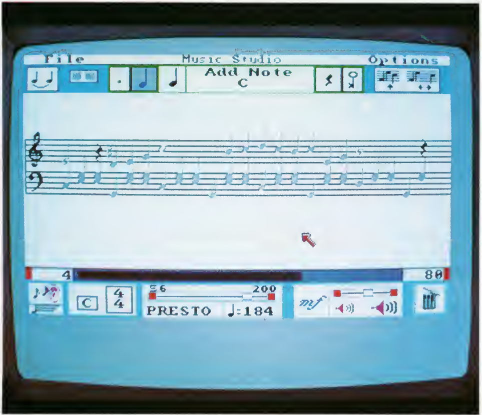
AMIGA



AMIGA

na, para efectuar modificaciones en la misma, así como posicionarse en un punto cualquiera de ésta, valiéndose del puntero analógico que recorre la barra horizontal, en la parte inferior de la pantalla, y el indicador numérico del extremo izquierdo de ésta, el cual varía mientras se realiza el posicionamiento.

Del mismo modo, pueden suprimirse notas innecesarias, o dar al traste con la composición completa, eliminándola de la memoria del ordenador para permitir la creación de una nueva, o la carga desde el soporte magnético de cualquier otra.



AMIGA





Elección de instrumentos

El programa está dotado de quince instrumentos diferentes predefinidos, y representados como colores en una «paleta» situada en la pantalla principal de composición. De esta forma, podemos indicar las notas que han de tocarse por cada uno de los instrumentos, con solo cambiar su color antes de introducirlas en el pentagrama.

En una composición sencilla puede emplearse un único instrumento. Sin embargo, el programa permite incluir más, incluso los quince, en el desarrollo de ésta, creando de esta forma un efecto polifónico espectacular.

Además, está siempre disponible la opción de alterar las cualidades sonoras de cualquiera de los instrumentos predefinidos, para conseguir unas características peculiares adecuadas a una composición determinada, lo que significa algo parecido a definir un nuevo instrumento inédito en la composición habitual.

La composición con la «caja de pinturas»

Este método de composición sustituye el empleo de la notación musical convencional, con la utilización de rectángulos de color, cuyo tamaño determina la duración de la nota, y su color qué instrumento debe tocarla. Si se desea, puede emplearse este sencillo método de composición, ya que el programa se encargará de traducirlo a la notación estándar, al acceder al Menú Principal de Composición.

Existen cinco tamaños diferentes de rectángulos, representando a las duraciones: entera, media, cuarto, etc...; tomando el programa por omisión el ancho del cuarto de nota. Por otra parte, la selección del instrumento se hace de la misma forma que la comentada anteriormente, a base de la paleta de colores.

Disponibilidad del programa

El programa se comercializa tanto para Commodore 64 como 128, además de en el Commodore Amiga, notablemente ampliado y mejorado, dadas las mejores prestaciones sonoras de este último modelo.

Caredit

Si alguna vez has deseado modificar el juego de caracteres del 64, seguramente habrás deseado también librarte de la maraña de números que ello conlleva. Para satisfacer tus deseos (sí, no lo niegues...), aquí tenemos a CAREDIT.

Coredit es un completo editor de carocteres poro el C-64. No sólo podrós diseñor con comodidad tus nuevos coracteres, sino que además obtendrós las líneos **DATA** poro incluir en tu propio programa, con las líneas **READ** necesarias... ¿Qué mós se puede pedir? No, los millones de dólares oún no sobemos progromorlos.

Muchas operaciones disponibles con sólo pulsar una tecla

Al ejecutar CAREDIT, tras una breve espera, aporecerá la pontallo de trobojo. Lo primero operoción que debemos reolizor es indicor el número del carácter que deseomos modificor. Este número lo podéis encontror en el monuol, y es el código de pontollo del carácter (ver página 132 del monuol).

Uno vez introducido, el corócter se copiará en la rejilla de puntos de lo porte superior de lo pontollo. En esta rejilla es en la que trobojoremos; para desplazarnos por ella, emplearemos un joystick conectodo al port 2.

Para activor o desactivar un punto del carácter, pulsemos el botón de disparo. Si el punto en el cual estomos estaba activado, se desoctivará, y viceverso.

Además de estos operaciones, la pulsación de una de las teclas de función nos dará acceso a diferentes opciones, que os enumeromos:

— F1: Esta tecla nos facilita lo operoción en modo editor (el que hemos descrito onteriormente) o en modo visor. Este último modo nos permite colocor el corócter con el que estomos trobojando en la porte inferior de lo pontollo (zono de visuolizoción). De esta forma, podemos ver el corócter con el que trabajamos, para poder opreciar el resultado.

La utilidad de este modo aparece, sobre todo, cuondo editomos corocteres que deben formar parte de un conjunto, por ejemplo, una gran nave espacial. Para ello, podemos ir seleccionando cada carócter, colocarlo en su posición en este óreo y ver el resultado.

Para movernos en esto zono, usoremos el joystick. El disparo depositará en la posición del cursor el coracter con el que octuolmente estemos trabajondo. Para volver al modo editor, pulsemos de nuevo F1. Recordamos que ninguna de las demás funciones son accesibles desde el modo visor, únicamente desde el editor.

— F2: Al pulsar esta teclo, se generarón automáticamente las líneas **DATA** correspondientes ol carócter actual. El programa comenzoró después desde el principio.

— F3: Al accionarla, podremos combiar el carácter con el que estemos trabajando, pasando a editar uno nuevo.

— F4: Esta es una tecla que deberá activarse CON MUCHO CUIDADO: Destruye el programa editor, y deja en memoria las



líneas **READ** y **DATA** que hemos generado... Y alguna sorpresilla más.

Como el editor se autodestruye (no en cinco segundos, pero si al ataque del F4), huelga decir que debemos grabar el programa antes de ejecutarlo por primera vez.

— F5: Esta opción nos permite obtener el negativo del carácter con el que estemos trabajando, de modo similar a un < RVS ON >.

— F6: Borra el carácter con el que estemos trabajando.

— F7: Invierte el carácter, colocándolo «cabeza abajo» (si un caracter tiene cabeza).

— F8: Copia un carácter determinado en el de trabajo. Al activar la opción, se nos preguntará el número del carácter a copiar. Esta opción es útil cuando deseamos crear varios caracteres muy similares; podemos crear el primero, copiarlo en otro y sobre él realizar las modificaciones, copiar y... Combinando esto con la inversión y la simetría, los efectos pueden ser sorprendentes.

— S: No una tecla de función, la S activa la Simetría. Nuestro carácter se invierte lateralmente, como visto desde la otra parte de la pantalla.

Vaya lío, ¿no? Tranquilos; todas las opciones aparecen en pantalla, por lo que no es necesario recordar nada.

Una advertencia: El programa lleva una pequeña rutina en código máquina, cuya misión es permitir que los nuevos caracteres sean visibles sólo en la parte inferior de la pantalla.

Si por cualquier razón (léase por una gracia del vecino, envidioso de nuestra habilidad) se interrumpe el programa, JAMAS debemos ejecutar **RUN,** pues además de la pérdida de los caracteres que pudiese haber, el ordenador se quedará «colgado», y habrá de ser apagado y encendido. Por tanto, para estas cosas, debemos proseguir con **RUN 1005.**

Nada más. Que disfrutéis con esos tecladas. Y, por favor, dejadnos ver en vuestros programas el resultado de vuestro arte.

LISTADO

```
1 GOTO1000                                          -075-
2 REM  ************************                     -144-
3 REM *         CAREDIT II        *                 -057-
4 REM *  (EDITOR DE CARACTERES)  *                  -202-
5 REM * (C) 1986 BY SINDIN SOFT *                   -117-
6 REM *   TU  MICRO  COMMODORE   *                  -150-
7 REM  ************************                     -149-
10 READA:IFA=999THENSTOP                            -069-
15 FORI=.TO7:READB:POKE14336+I+8*A,B:NEXT:GOTO10
                                                    -250-
990 PRINT"{CLR}{2 ABJ}1"CHR$(13):SN=1:GOSUB1500:PRINT
LN"DATA [3 9]"CHR$(13)                              -056-
991 PRINT"990"CHR$(13)"991"CHR$(13)"992"CHR$(13)
                                                    -039-
992 PRINT"GOTO998{HOM}";:POKE198,9:FORI=631TO640:POKE
I,13:NEXT:END                                       -128-
998 D=PEEK(61)+256*PEEK(62)+3:POKE252,D/256:POKE251,D
-256*PEEK(252)                                      -008-
999 POKED-2,.:POKED-1,.:POKE45,PEEK(251):POKE46,PEEK(
252):POKE828,.:SYS828                               -182-
1000 GOSUB1355                                      -047-
1005 M=1:E=1:PRINT"{WHT}{CLR}"                      -049-
1010 PRINT"{CLR}{ABJ}{GRN}{ROF}";                   -066-
1015 FORI=1TO8:PRINT"!8 O!<H>":NEXT                 -016-
1020 PRINT"<8 Y>{HOM}{15 ABJ}!6 *!VISUALIZACION!*!DE!
*!CARACTERES!7 *!";                                 -064-
1025 PRINT"{HOM}{ABJ}"SPC(20)"{RON}F1{ROF} : MODO
                                                    -092-
1030 PRINTSPC(20)"{RON}F2{ROF} : LISTAR DATAS
                                                    -030-
1035 PRINTSPC(20)"{RON}F3{ROF} : NUEVO CARACTER
                                                    -186-
1040 PRINTSPC(20)"{RON}F4{ROF} : KILL EDITOR
                                                    -216-
1045 PRINTSPC(20)"{RON}F5{ROF} : NEGATIVO           -040-
1050 PRINTSPC(20)"{RON}F6{ROF} : BORRAR             -144-
1055 PRINTSPC(20)"{RON}F7{ROF} : INVERTIR           -065-
1060 PRINTSPC(20)"{RON}F8{ROF} : TOMAR CARACTER
                                                    -179-
1062 PRINTSPC(20)"{RON}S {ROF} : SIMETRIA           -032-
1065 PRINT"{HOM}{10 ABJ}{GRN}MODO:";:IFM=.THENPRINT"V
ISOR "                                              -112-
```

```
1070 IFM=1THENPRINT"EDITOR"                              -249-
1075 ONM+1GOTO1295,1085                                  -055-
1080 REM MODO EDITOR                                     -003-
1085 X=.:Y=.:XV=.:YV=.:POKE198,.:GOSUB1185               -068-
1090 GETA$:IFA$="{F1 }"THENM=.:POKE1064+X+40*Y,CO:GOT
O1065                                                    -016-
1095 IFA$="{F2 }"THEN1285                                -236-
1100 IFA$="{F6 }"THENFORI=.TO7:V(I)=.:NEXT:GOSUB1205:
X=.:Y=.:XV=.:XY=.                                        -186-
1105 IFA$="{F3 }"THEN1085                                -223-
1110 IFA$="{F4 }"THEN1290                                -221-
1115 IFA$="{F5 }"THENFORI=.TO7:V(I)=255-V(I)::NEXT:GO
SUB1205:X=.:Y=.:XV=.:XY=.                                -129-
1120 IFA$="{F7 }"THENFORI=.TO7:A(I)=V(7-I):NEXT:GOSUB
1240:GOSUB1205:X=.:Y=.:XV=.:XY=.                         -075-
1125 IFA$="{F8 }"THENGOSUB1245                           -101-
1127 IFA$="S"THENGOSUB1450:GOSUB1205:X=.:Y=.:XV=.:YV=
.                                                        -080-
1130 I=PEEK(56320):X=X-((IAND8)=.)+((IAND4)=.):IFX>7T
HENX=.                                                   -166-
1135 IFX<.THENX=7                                        -054-
1140 Y=Y-((IAND2)=.)+((IAND1)=.):IFY>7THENY=.
                                                         -108-
1145 IFY<.THENY=7                                        -057-
1150 IFX<>XVORY<>YVTHENPOKE1064+XV+40*YV,CO:CO=PEEK(1
064+X+40*Y):XV=X:YV=Y:E=.                                -006-
1155 IF(PEEK(56320)AND16)=.THENIFE=.THENE=1:GOSUB1175
                                                         -191-
1160 IF(PEEK(56320)AND16)<>.THENIFE=1THENE=.
                                                         -112-
1165 POKE1064+X+40*Y,102                                 -059-
1170 GOTO1090                                            -236-
1175 IFV(Y)AND(2^(7-X))THENV(Y)=V(Y)AND(255-2^(7-X)):
CO=79:GOTO1225                                           -234-
1180 V(Y)=V(Y)OR(2^(7-X)):CO=207:GOTO1225                -149-
1185 PRINT"{HOM}{12 ABJ}CODIGO DEL CARACTER";:INPUTC
                                                         -246-
1190 IFC>255ORC<.THEN1185                                -227-
1195 PRINT"{HOM}{12 ABJ}CARACTER NUMERO "C"{IZQ}[4 ES
P]"                                                      -159-
1200 FORI=.TO7:V(I)=PEEK(14336+8*C+I):NEXT               -243-
1205 PRINT"{HOM}{ABJ}";:FORI=.TO7:FORJ=7TO.STEP-1
                                                         -017-
1210 IFV(I)AND(2^J)THENPRINT"{RON}|O|{ROF}";:GOTO1220
                                                         -071-
1215 PRINT"|O|";                                         -196-
1220 NEXT:PRINT"{DCH}":NEXT:CO=PEEK(1064)                -015-
1225 PRINT"{HOM}{ABJ}";:FORI=.TO7:PRINT"{9 DCH}";:J=V
(I):GOSUB1270:PRINTA$                                    -117-
1230 POKE14336+8*C+I,V(I):NEXT                           -239-
1235 RETURN                                              -203-
1240 FORI=.TO7:V(I)=A(I):NEXT:RETURN                     -049-
1245 PRINT"{HOM}{12 ABJ}CARACTER A COPIAR[2 ESP]";:IN
PUTC1                                                    -217-
1250 IFC1>255ORC1<.THEN1245                              -063-
1255 PRINT"{HOM}{12 ABJ}CARACTER NUMERO "C"{IZQ}[11 E
SP]"                                                     -124-
1260 FORI=.TO7:V(I)=PEEK(14336+8*C1+I):NEXT:GOSUB1205
:X=.:Y=.:YV=.:XV=.                                       -078-
1265 RETURN                                              -206-
1270 K=INT((JAND240)/16):A$=CHR$(48+K-7*(K>9))
```

```
                                                          -249-
1275 K=JAND15:A$=A$+CHR$(48+K-7*(K>9))                     -059-
1280 RETURN                                               -203-
1285 GOTO1500                                             -239-
1290 GOTO990                                              -199-
1295 XC=.:YC=.:XV=.:YV=.:R=1664:CA=PEEK(R):POKE198,.
                                                          -161-
1300 POKER+XC+40*YC,102                                   -063-
1305 GETA$:I=PEEK(56320)                                  -100-
1310 IFA$="{F1 }"THENPOKER+XC+40*YC,CA:X=.:Y=.:XV=.:Y
V=.:PRINT"{HOM}{10 ABJ}{5 DCH}";                           -144-
1315 IFA$="{F1 }"THENPRINT"EDITOR":CO=PEEK(1064):GOTO
1090                                                      -050-
1320 XC=XC-((IAND8)=.)+((IAND4)=.):IFXC>39THENXC=.
                                                          -178-
1330 YC=YC-((IAND2)=.)+((IAND1)=.):IFYC>8THENYC=.
                                                          -122-
1332 IFYC<.THENYC=8                                       -190-
1335 IFYC<>YVORXC<>XVTHENPOKER+XV+40*YV,CA:XV=XC:YV=Y
C:CA=PEEK(R+XC+40*YC)                                     -165-
1340 IF(IAND16)=.THENCA=C                                 -233-
1345 POKER+XC+40*YC,102:GOTO1305                          -132-
1350 END                                                  -192-
1355 REM INICIALIZACION ****************                  -184-
1360 PRINT"{GRN}{CLR}[4 ESP]ESPERE, POR FAVOR; INICIA
LIZANDO.[2 ESP]":POKE53280,.:POKE53281,.                   -157-
1365 DATA 120,169,0,141,14,220,169,36,141,20,3,169,19
2,141,21,3                                                -075-
1370 DATA 173,17,208,41,127,141,17,208,169,178,141,18
,208,169,1,141                                            -045-
1375 DATA 26,208,88,96,173,24,208,41,241,9,14,141,24,
208,169,70                                                -113-
1380 DATA 160,192,141,20,3,140,21,3,169,252,141,18,20
8,173,25,208                                              -177-
1385 DATA 141,25,208,76,188,254,173,24,208,41,241,9,4
,141,24,208                                               -153-
1390 DATA 169,36,160,192,141,20,3,140,21,3,169,178,14
1,18,208,173                                              -193-
1395 DATA 25,208,141,25,208,76,49,234                     -107-
1400 FORI=49152TO49255:READA:POKEI,A:NEXT:POKE56,56:P
OKE52,56                                                  -004-
1405 POKE56334,PEEK(56334)AND254:POKE1,PEEK(1)AND251:
J=38912                                                   -211-
1410 FORI=14336TO16383:POKEI,PEEK(I+J):NEXT               -135-
1415 POKE1,PEEK(1)OR4:POKE56334,PEEK(56334)OR1:SYS491
52:RETURN                                                 -050-
1450 PRINT"{HOM}ESPERE."                                  -192-
1452 FORJ=.TO7:FORI=.TO3:A(I)=-((V(J)AND(2^(7-I)))<>.
)                                                         -121-
1455 A(7-I)=-((V(J)AND(2^I))<>.):NEXT:V(J)=.:FORI=.TO
7:V(J)=V(J)+A(I)*(2^I)                                    -065-
1460 NEXT:NEXT:PRINT"{HOM}[7 ESP]":RETURN                 -187-
1500 LN=20:IFSNTHENRETURN                                 -152-
1510 PRINT"{CLR}{3 ABJ}{BLK}"LN"DATA";:PRINTMID$(STR$
(C),2);:FORI=.TO7                                         -080-
1520 PRINT","MID$(STR$(V(I)),2);:NEXT                     -164-
1530 PRINTCHR$(13)"1500 LN="MID$(STR$(LN+5),2)":IFSNT
!H!RE!T!"CHR$(13)"GOTO1005                                -078-
1540 POKE198,9:FORI=631TO640:POKEI,13:NEXT:PRINT"{HOM
}";:END                                                   -195-
```

HOT LINE

Nuestro amigo Sergio nos dirige una carta desde Argentina y nos cuenta una serie de problemas que tuvo al comprar su C-128.

Hemos de aclarar que todos estos problemas son debidos a un malo o nulo servicio de distribución en Argentina, y que los posibles compradores Europeos no deben tenerlos presentes; son dificultades de incompatibilidad de las normas americanas con las europeas.

En primer lugar está el problema de la alimentación, 110 V en lugar de los 220 V, la solución fue un transformador adecuado.

En segundo lugar está la ausencia de garantía, lo cual demuestra la inexistencia del servicio de distribución antes mencionado.

Otro problema surge a la hora de conectar el ordenador al televisor, como viene de origen preparado para sistema NTSC y no a PAL, nuestro amigo Sergio tuvo que convertir su televisor a bi-norma (80 dólares). En Europa, lo que se convierte es el ordenador y no la televisión, pero ante la inexistencia de servicio técnico la solución que adoptaste no es la mejor pero sí la única.

Tu último problema, aquí es idéntico, se trata de los manuales que vienen en inglés.

Nos alegramos mucho que te haya gustado nuestra revista, pero el desfase de números existente entre España y Argentina hará que esta contestación la recibas un poco tarde.

Ignoramos si en España hay algún programa de cálculo de estructuras para 64 ó 128, ya que es un tema muy específico, de todas formas creemos que, aunque no sea en España, algo habrá por ahí, debido a la gran cantidad de software disponible, sobre todo para C-64.

Por otra parte, el sistema FUERA ERRORES, todavía no está disponible para el C-128, debido a que el parque de ellos en España está muy por debajo del de C-64.

El programa que has hecho sobre funciones para C-128 envianoslo y entrará a formar parte de algún concurso de los que hacemos periódicamente.

Muchas gracias por escribir desde esa bonita tierra, no dudes en volver a hacerlo si tienes algún tipo de duda o problema.

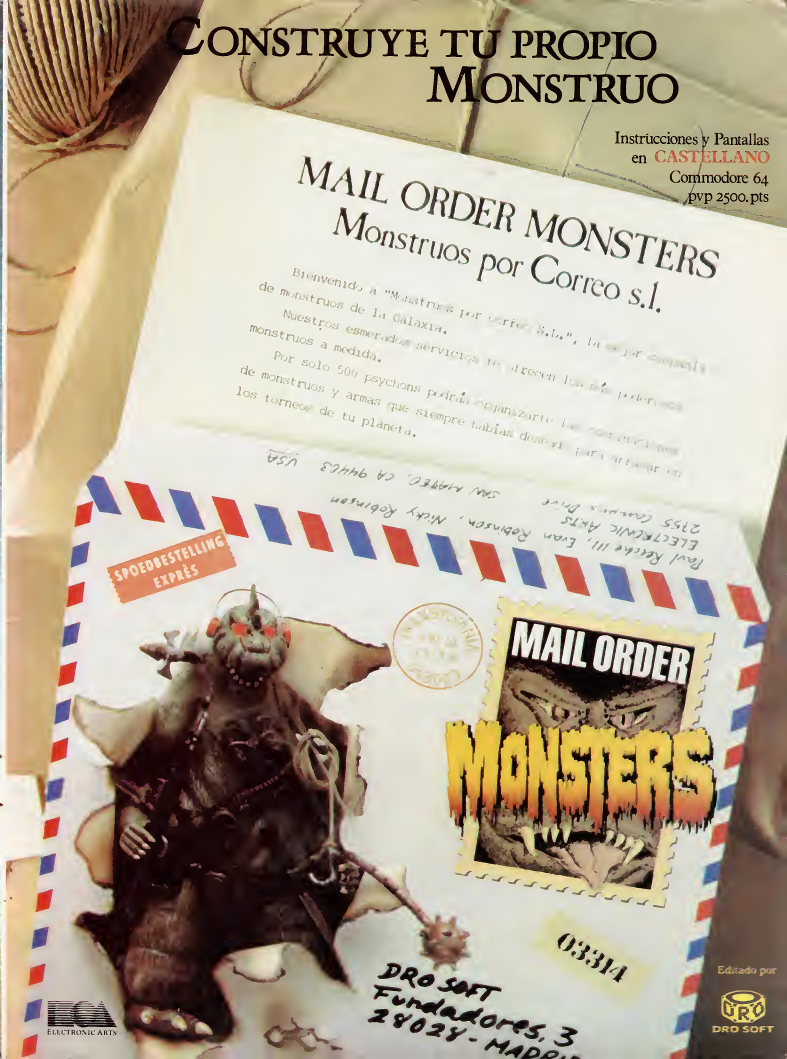
Cuatro problemas diferentes nos plantea un grupo de amigos de Hospitalet de Llobregat.

El primero de ellos es un poco extraño de comprender, ya que nos decís que habéis «tecleado» el programa de Vídeo Basic número 11 que ejecuta un hombrecillo con la tecla « », y el ordenador responde con READY. Este programa funciona perfectamente, pero al decir «tecleado», no aclaráis si habéis tecleado un trozo del programa, o si os referís al pseudo-programa que se genera desde otro; en el primer caso, debéis de haber copiado un fragmento no válido de programa (hay dos partes, una de chequeo y otra que es la que se ejecuta verdaderamente); en cuanto al segundo caso, el programa funciona bien, si ocurre esto, enviadnos la cinta original de Video Basic y os la cambiaremos por otra que funcione.

El segundo problema se refiere a un error en la línea 30 del programa «Fichero gráfico»; el error ILLEGAL QUANTITY, se producirá por un valor de la variable A mayor de 255. No decís si habéis utilizado el sistema FUERA ERRORES, ya que el programa funciona bien, si hay algún error en su copiado, pueden aparecer problemas de este tipo. Por favor, emplead FUERA ERRORES para evitar esto al máximo.

El tercer problema es que la línea 12 del programa «Copión-64» no entra en memoria y se borra automáticamente. De nuevo os remitimos a la respuesta del apartado anterior; utilizad el FUERA ERRORES y proceded así: apagad y encended el ordenador, cargad y ejecutad el programa fuera errores, teclead el programa de la revista comprobando que el último número del listado (entre guiones y que NO SE TECLEA), coincide al introducir la línea con RETURN, con el que aparecerá en la parte superior de la pantalla (en reverso y entre guiones también).

El cuarto problema es típico de un desajuste de la cabeza del Datassette o de cabezas sucias. Para remediarlo limpiad las cabezas del Datassette con un algodón impregnado en alcohol, dejad un par de horas hasta que seque y probad si la situación ha mejorado; si no es así, habrá que proceder a un ajuste de cabezales, para lo cual, lo mejor es comprar un KIT de ajuste y seguir sus instrucciones; suele ser un programa que indica el estado de la cabeza y un manual que explica lo que hay que hacer. Básicamente se trata de girar con un destornillador muy fino que se introduce por el orificio adecuado, un tornillito de ajuste. Pero no lo hagáis de cualquier manera, pues los resultados pueden ser aún peores.



CONSTRUYE TU PROPIO MONSTRUO
Instrucciones y Pantallas en CASTELLANO
Commodore 64
pvp 2500.pts
MAIL ORDER MONSTERS
Monstruos por Correo s.l.
Bienvenido a "Monstruos por correo S.L.", la mejor compañía
de monstruos de la Galaxia.
Nuestros esmerados servicios te ofrecen los más poderosos
monstruos a medida.
Por solo 500 psychons podrás organizarte las
de monstruos y armas que siempre habías deseado para
los torneos de tu planeta.
Paul Reiche III, Evan Robinson, Nicky Robinson
ELECTRONIC ARTS
2155 Campus Drive
SAN MATEO, CA 94403 USA
SPOEDBESTELLING EXPRÈS
MAIL ORDER MONSTERS
03314
DRO SOFT
Fundadores, 3
28028 - MADRID
ELECTRONIC ARTS
Editado por
DRO SOFT



SPECTRUM - COMMODORE - AMSTRAD
DEACTIVATORS
Instrucciones y Pantallas en
CASTELLANO
¡Mal momento amigos! Como Jefe de Seguridad del Instituto de Investigaciones Gravitacionales, estaba tranquilamente apoltronado en mi despacho cuando mi "teléfono caliente" sonó. Un grupo de saboteadores fanáticos se ha infiltrado en el Instituto y ha plantado numerosas bombas de relojería y ADEMAS ha reprogramado mis robots guardianes para darme el susto de mi vida.
Bien, ¡esto es todo! He tenido que llamar a mis fuerzas de élite, los Androides Desactivadores, para limpiar los cinco laboratorios y deshacerse de los desapacibles guardianes tontos. Por supuesto que no es tan fácil moverse por entre habitaciones cuyos suelos sean techos y los techos puedan ser paredes. Si los DEACTIVATORS pueden arrojar todas las bombas fuera del Centro antes de que todos desafiemos las leyes de la gravedad, es posible que pueda continuar mi siesta.
Editado por DRO SOFT
Fundadores, 3 28028 MADRID
Tlfs.: 255 45 00/09
DRO SOFT